

**PENGEMBANGAN APLIKASI *MOBILE AUGMENTED REALITY*
SEBAGAI MEDIA BELAJAR PENGENALAN DASAR HURUF
HIJAIYAH**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Adji Reno Muhammad

NIM: 135150200111018



PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018

PENGESAHAN

PENGEMBANGAN APLIKASI *MOBILE AUGMENTED REALITY* SEBAGAI MEDIA
BELAJAR PENGENALAN DASAR HURUF

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Adji Reno Muhammad
NIM: 135150200111018

Skripsi ini telah diuji dan dinyatakan lulus pada
2 Agustus 2018
Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I



Wibisono Sukmo Wardhono, S.T, M.T

NIK: 201008 820404 1 001

Dosen Pembimbing II



Tri Afianto, S.T, M.T

NIK: 201309 851213 1 001

Mengetahui
Ketua Jurusan Teknik Informatika




Tri Astoto Kurniawan, S.T, M.T, Ph.D

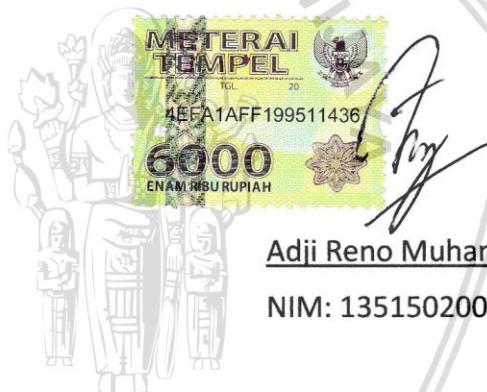
NIP: 19710518 200312 1 001

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 10 Agustus 2018



Adji Reno Muhammad

NIM: 135150200111018

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa, karena berkat rahmat dan hidayah-Nya, penulis dapat menyelesaikan dokumen skripsi yang berjudul “Pengembangan Aplikasi *Mobile Augmented Reality* Sebagai Media Belajar Pengenalan Dasar Huruf Hijaiyah”.

Skripsi merupakan salah satu mata kuliah yang wajib ditempuh sebagai syarat kelulusan di Fakultas Ilmu Komputer Universitas Brawijaya. Dalam penulisan dokumen ini, penulis mendapatkan banyak bantuan dari berbagai pihak baik secara moril maupun materil. Dalam kesempatan ini penulis ingin mengucapkan terimakasih yang sebesar-besarnya kepada:

1. Bapak Wibisono Sukmo Wardhono, S.T, M.T, selaku dosen pembimbing 1 dan Bapak Tri Afirianto, S.T, M.T, selaku dosen pembimbing 2 yang telah membimbing, memberi saran dan memotivasi penulis selama penyusunan skripsi.
2. Kedua orang tua penulis, yang selalu membantu dan memberikan dukungan selama penulis menempuh studi di Universitas Brawijaya.
3. Seluruh dosen dan civitas akademika Fakultas Ilmu Komputer, yang memberikan wawasan, bantuan dan dukungan yang sangat bermanfaat bagi penulis.
4. Teman-teman dari kontrakan KK 12c yang telah banyak membantu kelancaran pelaksanaan dan penyusunan dokumen skripsi ini.

Penulis menyadari bahwa dalam penyusunan dokumen skripsi ini masih banyak kekurangan baik dari isi maupun susunannya. Oleh karena itu, saya sebagai penulis dengan senang hati akan menerima kritik dan sarang yang membangun sebagai acuan untuk perbaikan kedepannya.

Malang, 10 Agustus 2018

Penulis

Djiiren.adjireno@gmail.com.com

ABSTRAK

Pengenalan dan penguasaan huruf hijaiyah sejak dini sangatlah penting, karena huruf hijaiyah merupakan dasar untuk membaca dan mempelajari kitab suci Al-Qur'an. Namun dalam pengucapannya, banyak orang yang masih salah dalam mengucapkan beberapa huruf hijaiyah yang kemungkinan disebabkan oleh bahasa ibu dan faktor kebiasaan. Umumnya huruf hijaiyah dikenalkan kepada anak saat mereka memasuki taman kanak-kanak atau sekolah dasar. Dalam proses belajar, terkadang ada anak yang memiliki sifat pemalu yang disebabkan oleh kurang beraninya anak ketika berbicara dengan guru atau orang dewasa lain, tidak dapat menatap mata orang lain ketika berbicara, anak tidak banyak bicara, atau anak kurang terbuka. Ditambah lagi ada beberapa anak yang cenderung belajar menggunakan visual, sementara yang lain cenderung menggunakan audio ataupun kinestetik. Dari permasalahan tersebut, dikembangkanlah aplikasi *mobile augmented reality* berbasis Android sebagai media belajar pengenalan dasar huruf hijaiyah, yang memiliki fitur untuk menampilkan varian huruf hijaiyah, animasi penulisan huruf hijaiyah, suara pengucapan huruf hijaiyah, dan transliterasi dari huruf hijaiyah. Aplikasi ini dikembangkan dengan metode *waterfall*. Pengujian yang dilakukan adalah pengujian validasi, pengujian kondisi optimal *marker* berdasarkan besar dan jarak, serta pengujian aplikasi. Pengujian validasi memberikan hasil bahwa sistem valid. Pengujian kondisi optimal *marker* memberikan hasil bahwa kondisi optimal *marker* ukuran 2,5x2,5cm adalah 4-8cm, ukuran 5x5cm adalah 8-17cm, ukuran 7,5x7,5cm adalah 12-26cm, ukuran 10x10 adalah 16-34cm, dan ukuran 12,5x12,5cm adalah 19-42cm. Pengujian aplikasi memberikan hasil bahwa dengan menggunakan aplikasi dapat meningkatkan nilai siswa sebesar 11,19%.

Kata kunci: *mobile, aplikasi, augmented reality, media pembelajaran, huruf hijaiyah*

ABSTRACT

The introduction and mastery of hijaiyah letters from an early age is very important, because the hijaiyah is the basis for reading and studying the holy book Al-Qur'an. But in pronunciation, many people are still wrong in pronouncing some letters hijaiyah which may be caused by the mother language and habitual factors. Generally, hijaiyah letters are introduced to children as they enter kindergarten or primary school. In the learning process, sometimes there are children who have a shy nature caused by lack of bravery of children when talking to teachers or other adults, not being able to look into other people's eyes when talking, the child does not talk much, or the child is less open. Plus there are some children who tend to learn to use visual, while others tend to use audio or kinesthetic. From these problems, Android-based augmented reality mobile application developed as a learning media introduction of basic hijaiyah letters, which features to display the variants of hijaiyah letters, writing animation of letters hijaiyah, hijaiyah letter pronunciation, and transliteration of hijaiyah letters. This application is developed by waterfall method. The test is validation testing, optimum condition of marker based on size and distance, and application testing. Validation testing gives results that the system is valid. Testing the optimal marker conditions gives the results that the optimal condition of the marker size 2,5x2,5cm is 4-8cm, the size of 5x5cm is 8-17cm, the size of 7,5x7,5cm is 12-26cm, the size of 10x10 is 16-34cm, and the size is 12,5x12,5cm is 19-42cm. Application testing gives results that by using the application can improve student scores by 11.19%.

Keywords: *mobile, application, augmented reality, learning media, hijaiyah letters.*

DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	v
ABSTRACT	vi
DAFTAR ISI	vii
DAFTAR TABEL.....	x
DAFTAR GAMBAR.....	xii
DAFTAR PSEUDOCODE	xiv
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan	2
1.4 Manfaat.....	2
1.5 Batasan Masalah.....	3
1.6 Sistematika Pembahasan.....	3
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Media Pembelajaran.....	5
2.2 Augmented Reality	5
2.3 Huruf Hijaiyah	7
2.3.1 Konsonan.....	7
2.3.2 Vokal.....	8
2.4 Aplikasi Pendukung.....	9
2.4.1 Unity	9
2.4.2 Vuforia SDK.....	10
2.4.3 Blender 3D.....	10
2.5 Metode <i>Waterfall</i>	11
2.6 UML (<i>Unified Modeling Language</i>).....	12
2.6.1 <i>Use case diagram</i>	12
2.6.2 <i>Activity diagram</i>	13

2.6.3 Class diagram	14
BAB 3 METODOLOGI PENELITIAN	16
3.1 Studi Pustaka.....	16
3.2 Analisis Kebutuhan (<i>Requirement Analysis and Definition</i>)	17
3.3 Perancangan (<i>System and Software Design</i>)	17
3.4 Implementasi (<i>Implementation and Unit Testing</i>)	17
3.5 Pengujian (<i>Integration and System Testing</i>).....	17
3.6 Penarikan Kesimpulan dan Saran	18
BAB 4 PERANCANGAN.....	19
4.1 Perancangan Umum Sistem.....	19
4.2 Analisis Kebutuhan Perangkat Lunak.....	19
4.2.1 Identifikasi aktor	19
4.2.2 Analisis kebutuhan fungsional	20
4.2.3 Analisis kebutuhan non fungsional	21
4.3 Perancangan Perangkat Lunak	21
4.3.1 Use case diagram	21
4.3.2 Activity diagram	28
4.3.3 Class diagram	32
4.3.4 Perancangan antarmuka	33
4.3.5 Perancangan marker	36
BAB 5 IMPLEMENTASI	38
5.1 Spesifikasi Lingkungan Implementasi	38
5.1.1 Spesifikasi perangkat keras	38
5.1.2 Spesifikasi perangkat lunak.....	39
5.2 Implementasi <i>Marker</i>	39
5.3 Implementasi Konten 3D	40
5.4 Implementasi Antarmuka Aplikasi.....	40
5.4.1 Menu utama.....	40
5.4.2 Camera view.....	41
5.4.3 Halaman bantuan.....	42
5.4.4 Halaman tentang.....	43
5.5 Implementasi Kode Program	43

5.5.1 Implementasi deteksi <i>marker</i>	43
5.5.2 Implementasi fitur varian.....	44
5.5.3 Implementasi fitur animasi	45
5.5.4 Implementasi fitur transliterasi	45
5.5.5 Implementasi fitur pemutaran suara	47
BAB 6 PENGUJIAN DAN ANALISIS.....	49
6.1 Pengujian	49
6.1.1 Pengujian validasi.....	49
6.1.2 Pengujian kondisi optimal <i>marker</i>	52
6.1.3 Pengujian aplikasi.....	52
6.2 Hasil dan Analisis	52
6.2.1 Hasil dan analisis pengujian validasi	52
6.2.2 Hasil dan analisis pengujian kondisi optimal <i>marker</i>	53
6.2.3 Hasil dan analisis pengujian aplikasi	55
BAB 7 PENUTUP	57
7.1 Kesimpulan.....	57
7.2 Saran	57
DAFTAR PUSTAKA.....	58

DAFTAR TABEL

Tabel 2.1 Daftar huruf bahasa Arab dan transliterasinya	7
Tabel 2.2 Vokal tunggal bahasa Arab	9
Tabel 2.3 Deskripsi simbol-simbol <i>use case diagram</i>	13
Tabel 2.4 Deskripsi simbol-simbol <i>activity diagram</i>	13
Tabel 2.5 Deskripsi simbol-simbol <i>class diagram</i>	14
Tabel 4.1 Identifikasi aktor	19
Tabel 4.2 Spesifikasi kebutuhan fungsional <i>user</i>	20
Tabel 4.3 Spesifikasi kebutuhan non fungsional	21
Tabel 4.4 Skenario <i>use case</i> melihat halaman bantuan	22
Tabel 4.5 Skenario <i>use case</i> melihat halaman tentang	22
Tabel 4.6 Skenario <i>use case</i> keluar aplikasi	23
Tabel 4.7 Skenario <i>use case</i> memulai mode AR	24
Tabel 4.8 Skenario <i>use case</i> lihat varian huruf	25
Tabel 4.9 Skenario <i>use case</i> lihat animasi penulisan	26
Tabel 4.10 Skenario <i>use case</i> lihat transliterasi	26
Tabel 4.11 Skenario <i>use case</i> dengar pembacaan huruf	27
Tabel 4.12 Deskripsi <i>class diagram</i>	33
Tabel 4.13 Penjelasan tombol pada halaman menu utama	34
Tabel 4.14 Penjelasan tombol pada saat <i>marker</i> teridentifikasi valid	34
Tabel 4.15 Penjelasan tombol pada fitur animasi dan suara	35
Tabel 4.16 Penjelasan tombol pada halaman bantuan	36
Tabel 5.1 Spesifikasi perangkat keras komputer	38
Tabel 5.2 Spesifikasi perangkat keras <i>smartphone</i>	38
Tabel 5.3 Spesifikasi perangkat lunak komputer	39
Tabel 5.4 Spesifikasi perangkat lunak <i>smartphone</i>	39
Tabel 6.1 Kasus uji melihat halaman bantuan	49
Tabel 6.2 Kasus uji melihat halaman tentang	49
Tabel 6.3 Kasus uji keluar aplikasi	50
Tabel 6.4 Kasus uji memulai mode AR	50
Tabel 6.5 Kasus uji lihat varian huruf	50

Tabel 6.6 Kasus uji lihat animasi penulisan.....	51
Tabel 6.7 Kasus uji lihat transliterasi.....	51
Tabel 6.8 Kasus uji dengar pembacaan huruf	52
Tabel 6.9 Hasil pengujian validasi	52
Tabel 6.10 Hasil pengujian kondisi optimal <i>marker</i>	53
Tabel 6.11 Hasil pengujian aplikasi	55



DAFTAR GAMBAR

Gambar 2.1 Representasi sederhana dari continuum RV.....	6
Gambar 2.2 <i>Marker</i> sederhana.....	6
Gambar 2.3 Unity <i>interface</i>	9
Gambar 2.4 Blender <i>interface</i>	11
Gambar 2.5 Contoh metode <i>waterfall</i>	11
Gambar 3.1 Diagram alir	16
Gambar 4.1 <i>Use case diagram</i>	21
Gambar 4.2 <i>Activity diagram</i> melihat halaman bantuan	28
Gambar 4.3 <i>Activity diagram</i> melihat halaman tentang	29
Gambar 4.4 <i>Activity diagram</i> keluar aplikasi	29
Gambar 4.5 <i>Activity diagram</i> memuliah mode AR	30
Gambar 4.6 <i>Activity diagram</i> lihat varian huruf	30
Gambar 4.7 <i>Activity diagram</i> lihat animasi penulisan	31
Gambar 4.8 <i>Activity diagram</i> lihat transliterasi	31
Gambar 4.9 <i>Activity diagram</i> dengar pembacaan huruf	32
Gambar 4.10 <i>Class diagram</i>	32
Gambar 4.11 Perancangan halaman awal / menu utama	33
Gambar 4.12 Perancangan halaman pada saat <i>marker</i> teridentifikasi valid	34
Gambar 4.13 Perancangan halaman pada fitur varian dan transliterasi.....	35
Gambar 4.14 Perancangan halaman bantuan	35
Gambar 4.15 Perancangan halaman tentang	36
Gambar 4.16 Gambar desain <i>marker</i>	37
Gambar 5.1 <i>Marker</i> dengan tanda pengenal.....	39
Gambar 5.2 Implementasi objek 3D	40
Gambar 5.3 Implementasi menu utama	40
Gambar 5.4 Tampilan saat <i>marker</i> valid	41
Gambar 5.5 Tampilan saat memilih fitur varian	41
Gambar 5.6 Tampilan saat memilih fitur transliterasi	42
Gambar 5.7 Implementasi halaman bantuan	42
Gambar 5.8 Implementasi halaman tentang	43

Gambar 6.1 Grafik hasil pengujian aplikasi.....	56
---	----



DAFTAR PSEUDOCODE

Pseudocode 5.1 Implementasi deteksi <i>marker</i>	43
Pseudocode 5.2 Implementasi fitur varian	44
Pseudocode 5.3 Implementasi fitur animasi	45
Pseudocode 5.4 Implementasi fitur transliterasi.....	46
Pseudocode 5.5 Implementasi fitur pemutaran suara	48



BAB 1 PENDAHULUAN

1.1 Latar Belakang

Pengenalan dan penguasaan huruf hijaiyah sejak dini sangatlah penting, karena huruf hijaiyah merupakan dasar untuk membaca dan mempelajari kitab suci Al-Qur'an. Namun dalam pengucapannya, banyak orang yang masih salah dalam mengucapkan beberapa huruf hijaiyah yang kemungkinan disebabkan oleh bahasa ibu dan faktor kebiasaan (Ifka, 2013). Apabila pada saat membaca huruf hijaiyah sudah terbiasa dengan pengucapan yang salah maka bisa menimbulkan arti yang berbeda sehingga kita dituntut untuk membaca secara baik dan benar.

Umumnya huruf hijaiyah dikenalkan kepada anak saat mereka memasuki taman kanak-kanak atau sekolah dasar. Dalam proses belajar, terkadang ada anak yang memiliki sifat pemalu yang disebabkan oleh kurang beraninya anak ketika berbicara dengan guru atau orang dewasa lain, tidak dapat menatap mata orang lain ketika berbicara, anak tidak banyak bicara, atau anak kurang terbuka (Dewi, 2005). Ditambah lagi ada beberapa anak yang cenderung belajar menggunakan visual, sementara yang lain cenderung menggunakan audio ataupun kinestetik. Karena gaya belajar yang berbeda ini, penting bagi pengajar untuk menggunakan kombinasi metode pengajaran dan membuat lingkungan belajar menjadi interaktif dan menstimulasi untuk mencapai hasil pembelajaran yang baik. Pengetahuan dan pemahaman yang lebih baik tentang gaya belajar mungkin juga menjadi penting seiring bertambahnya ukuran kelas dan seiring kemajuan teknologi terus menciptakan jenis siswa yang masuk ke pendidikan yang lebih tinggi (Gilakjani, 2012).

Beberapa penelitian telah dilakukan untuk membuktikan bahwa penggunaan media visual dapat membantu pelajar dalam proses belajarnya. Salah satunya adalah penelitian tentang penggunaan media visual yang digunakan untuk mengajar bahasa Mandarin siswa kelas 5 SD (Rochmani, 2011). Penelitian ini telah berhasil membuktikan bahwa dengan menggunakan metode pembelajaran kosakata melalui media visual dapat meningkatkan prestasi belajar Bahasa Mandarin siswa kelas 5 SD. Perbandingan yang dilakukan dilihat dari hasil tes tertulis maupun tes secara lisan dengan peningkatan dari rata-rata nilai 64 sebelum menggunakan media visual menjadi rata-rata 87,5 setelah menggunakan media visual.

Penelitian selanjutnya adalah penelitian tentang penggunaan *augmented reality* dengan materi pembelajaran cetak tradisional (Asai, et al., 2005). Sebanyak 22 mahasiswa yang memiliki pengelihan normal dan tidak memiliki pengalaman *augmented instructions* sebelumnya berpartisipasi dalam penelitian ini. Tugas yang diberikan kepada partisipan adalah membaca dokumen dua halaman dan melihat model 3D dengan dua sistem presentasi yaitu *handheld PC* dan *head-mounted display* (HMD). Pada penelitian ini, mereka membuktikan bahwa

penggunaan *augmented reality* dengan materi pembelajaran cetak tradisional memiliki potensi untuk memberi kita cara baru untuk menggunakan materi pembelajaran tercetak. Pada penelitian ini mereka juga menunjukkan bahwa *handheld PC* lebih sesuai daripada *HMD* sebagai sistem presentasi untuk penggunaan dalam jangka waktu yang lama.

Berdasarkan penjelasan tersebut penulis akan membangun sebuah aplikasi *mobile* berjudul “**Pengembangan Aplikasi Mobile Augmented Reality Sebagai Media Belajar Pengenalan Dasar Huruf Hijaiyah**”. *Augmented reality* akan diimplementasikan pada *smartphone* dengan sistem operasi Android. Aplikasi *augmented reality* akan menampilkan huruf hijaiyah, menampilkan animasi cara penulisan huruf hijaiyah, serta suara pengucapan huruf hijaiyah sehingga mempermudah anak untuk mengenal huruf hijaiyah.

1.2 Rumusan Masalah

Berdasarkan permasalahan yang telah disampaikan pada latar belakang, maka disusunlah rumusan masalah sebagai berikut:

1. Bagaimana mengimplementasikan aplikasi untuk pembelajaran huruf hijaiyah yang menarik dan mudah digunakan?
2. Bagaimana kondisi optimal aplikasi *mobile augmented reality* berdasarkan besar dan jarak dari *marker*?
3. Bagaimana peningkatan nilai membaca siswa setelah menggunakan aplikasi *mobile augmented reality* untuk pembelajaran huruf hijaiyah?

1.3 Tujuan

Berikut adalah tujuan dari penelitian ini:

1. Merancang aplikasi *mobile augmented reality* untuk media belajar pengenalan huruf hijaiyah.
2. Menguji kemampuan optimal dari aplikasi *mobile augmented reality* berdasarkan besar dan jarak dari *marker*.
3. Untuk mengukur peningkatan nilai membaca siswa setelah menggunakan aplikasi *mobile augmented reality* dalam pembelajaran huruf hijaiyah.

1.4 Manfaat

Manfaat dari penelitian ini yang dapat diambil adalah:

1. Bagi Penulis:
 - a. Dapat menerapkan ilmu yang didapat selama kuliah sehingga diharapkan berguna di masyarakat.
 - b. Dapat mengetahui konsep kerja dari *augmented reality* pada *smartphone*.

2. Bagi pengguna:
 - a. Dapat mempermudah belajar huruf hijaiyah.
 - b. Dapat membantu proses pembelajaran dalam mengenalkan huruf hijaiyah.
3. Bagi Instansi:
 - a. Dapat ikut serta berpartisipasi dalam pengembangan dan pemanfaatan teknologi IT dalam bidang pendidikan.

1.5 Batasan Masalah

Agar pembahasan tidak melebar dan lebih terperinci sehingga sesuai dengan apa yang diharapkan, maka diberikanlah pembatasan masalah sebagai berikut:

1. Aplikasi ini dibuat dengan menggunakan Unity versi 5.5.2f1 dengan *library* Vuforia.
2. Aplikasi dijalankan pada perangkat *mobile* bersistem operasi Android dengan minimal versi 4.0.
3. *Augmented reality* yang diterapkan menggunakan *marker*.
4. *Marker* yang digunakan untuk menampilkan objek adalah *marker* yang sudah dikenali sistem.

1.6 Sistematika Pembahasan

Bagian ini berisi kerangka pembahasan yang bertujuan untuk memberikan gambaran dan uraian dari penyusunan skripsi secara garis besar yang meliputi beberapa bab sebagai berikut:

BAB 1 : PENDAHULUAN

Bab ini menguraikan latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, serta manfaat perancangan aplikasi *mobile augmented reality* untuk pembelajaran huruf hijaiyah.

BAB 2 : LANDASAN KEPUSTAKAAN

Pada bab ini penulis akan menjelaskan tentang teori yang menjadi acuan dalam perancangan aplikasi *mobile augmented reality* untuk pembelajaran huruf hijaiyah.

BAB 3 : METODOLOGI PENELITIAN

Pada bab ini akan dijelaskan metodologi penelitian yang digunakan oleh penulis dalam melakukan penelitian aplikasi *mobile augmented reality* untuk pembelajaran huruf hijaiyah.

BAB 4 : PERANCANGAN

Pada bab ini akan menjelaskan bagian perancangan aplikasi mulai dari perancangan umum sistem, analisis kebutuhan perangkat lunak, dan perancangan perangkat lunak.

BAB 5 : IMPLEMENTASI

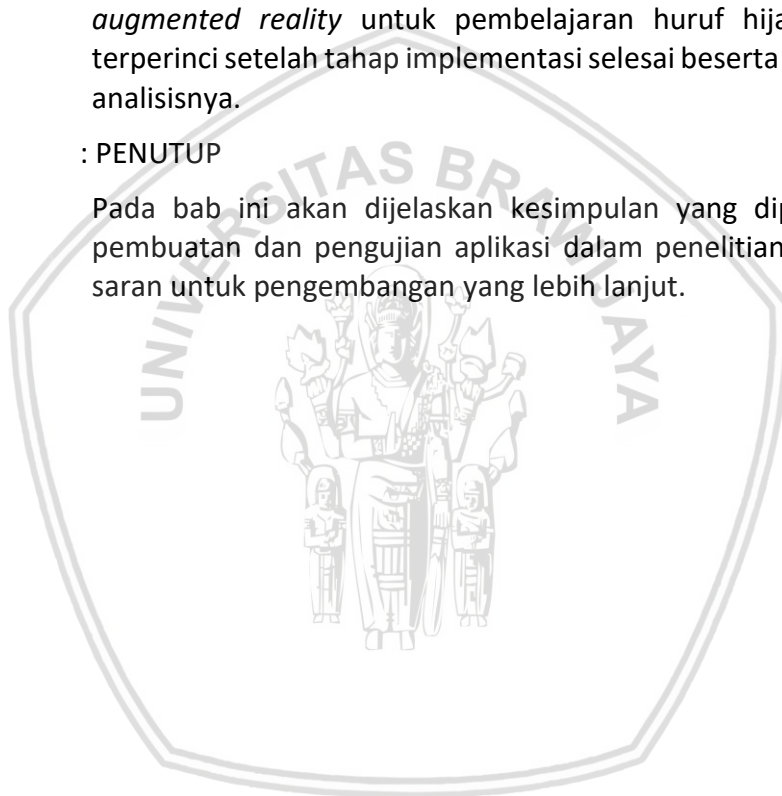
Pada bab ini akan dijelaskan proses implementasi yang akan dilakukan pada aplikasi *mobile augmented reality* untuk pembelajaran huruf hijaiyah setelah tahap perancangan selesai.

BAB 6 : PENGUJIAN DAN ANALISIS

Bab ini menjelaskan tentang proses pengujian aplikasi *mobile augmented reality* untuk pembelajaran huruf hijaiyah secara terperinci setelah tahap implementasi selesai beserta dengan hasil analisisnya.

BAB 7 : PENUTUP

Pada bab ini akan dijelaskan kesimpulan yang diperoleh dari pembuatan dan pengujian aplikasi dalam penelitian ini, beserta saran untuk pengembangan yang lebih lanjut.



BAB 2 LANDASAN KEPUSTAKAAN

Bab ini membahas tentang dasar-dasar teori yang diperlukan untuk penelitian. Teori yang dibahas yaitu penjelasan tentang media pembelajaran, *augmented reality*, huruf hijaiyah, unity, vuforia sdk, blender 3D, metode *waterfall*, dan UML (*Unified Modeling Language*).

2.1 Media Pembelajaran

Secara umum, media pembelajaran dapat diartikan sebagai alat bantu proses belajar dan mengajar. Sedangkan menurut Asyar (2012) media pembelajaran adalah segala sesuatu yang dapat menyalurkan atau menyampaikan pesan dari sumber secara terencana, membuat penerimanya dapat melakukan proses belajar secara efisien dan efektif, sehingga membuat lingkungan belajar yang kondusif.

Perkembangan zaman saat ini menunjukkan kecenderungan bahwa pada masa depan dalam konteks pendidikan, setiap individu akan dihadapkan dengan kebutuhan dan tantangan yang tidak sepenuhnya bisa dapat terpenuhi melalui sistem pendidikan yang konvensional (Sidiq, 2016). Dengan demikian, metode pembelajaran harus mulai diarahkan dari sebuah pembelajaran konvensional menjadi pembelajaran yang mengikuti perkembangan zaman.

Pembelajaran visual adalah pembelajaran yang menggunakan media visual sebagai perantara untuk menyampaikan pesannya. Media visual memegang peran yang penting dalam proses belajar mengajar. Penggunaan media visual dalam proses mengajar dapat mencegah terjadinya kesalahan penafsiran ketika siswa menerima informasi dari pengajar yang hanya menggunakan verbal dalam menyampaikannya. Penggunaan media visual dalam proses belajar dapat memperlancar pemahaman dan memperkuat ingatan. Penggunaan media visual dalam belajar juga dapat menumbuhkan minat siswa serta dapat memberikan dukungan antara isi materi pelajaran dengan dunia nyata.

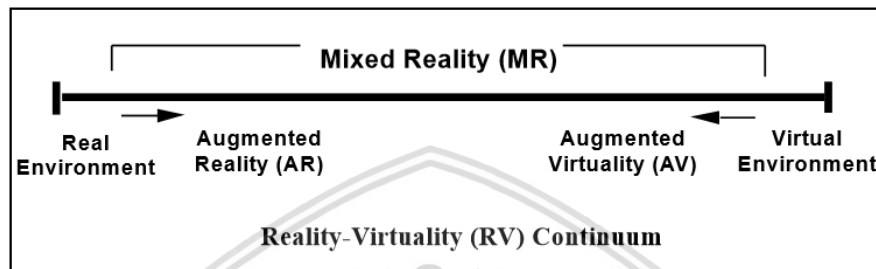
Beberapa penelitian telah dilakukan untuk membuktikan bahwa penggunaan media visual dapat membantu pelajar dalam proses belajarnya. Salah satunya adalah penelitian yang dilakukan oleh Rochmani (2011) yang membuktikan bahwa penggunaan media visual untuk pembelajaran kosakata dapat meningkatkan prestasi belajar bahasa Mandarin siswa SD kelas 5 dengan peningkatan dari rata-rata nilai 64 menjadi 87,5.

2.2 Augmented Reality

Augmented reality adalah sebuah teknologi yang memungkinkan penggunaanya untuk melihat dunia nyata dengan objek virtual yang digabungkan dengan dunia nyata secara *real time*, baik itu objek 2D ataupun 3D sehingga akan terlihat seolah-olah objek virtual dan objek nyata tersebut hidup berdampingan di tempat dan waktu yang sama (Azuma, 1997).

Menurut Azuma (1997) untuk menghindari pembatasan *augmented reality* terhadap teknologi tertentu, *augmented reality* didefinisikan sebagai sistem yang memiliki tiga karakteristik berikut:

1. Menggabungkan yang nyata dan yang virtual
2. Interaktif secara *real time*
3. Menampilkan objek dalam 3D

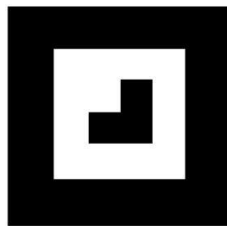


Gambar 2.1 Representasi sederhana dari continuum RV

Gambar 2.1 menunjukkan bahwa pada bagian kiri *continuum* merupakan lingkungan yang hanya terdiri dari objek nyata, termasuk apa pun yang dapat diamati saat melihat pemandangan dunia nyata baik secara langsung, melalui jendela, atau melalui tampilan berupa video. Sedangkan pada bagian kanan, mendefinisikan lingkungan yang terdiri dari objek virtual seperti simulasi grafik komputer baik berbasis monitor maupun *immersive* (Milgram, et al., 1994).

Dalam implementasinya, terdapat dua tipe *augmented reality*. Tipe pertama adalah tipe *marker based* yang menggunakan kamera dan penanda visual (*marker*), sedangkan yang kedua adalah tipe *markerless* yang menggunakan data posisi seperti GPS (*Global Positioning System*) dan kompas (Katiyar, et al., 2015).

Marker augmented reality adalah gambar yang dapat dideteksi oleh kamera yang nantinya digunakan dengan perangkat lunak sebagai lokasi untuk menampilkan aset virtualnya. Sebagian besar *marker* biasanya berwarna hitam dan putih. Warna lain bisa digunakan selama kontras diantara warnanya dapat dikenali dengan benar oleh kamera (Katiyar, et al., 2015). *Marker augmented reality* sederhana dapat terdiri dari satu atau lebih bentuk dasar yang terbuat dari kotak hitam dengan latar belakang putih. Gambar 2.2 adalah salah satu contoh dari *marker* sederhana.



Gambar 2.2 Marker sederhana

2.3 Huruf Hijaiyah

Huruf hijaiyah atau abjad Arab adalah hasil kodifikasi dari aksara bahasa Arab yang digunakan untuk penulisan bahasa Arab dan terdiri dari 29 huruf (Apriyani, et al., 2016). Dalam penulisannya banyak huruf hijaiyah yang memiliki bentuk serupa, yang membedakan adalah adanya titik diatas atau dibawah bentuk huruf. Titik tersebut merupakan bagian yang tidak terpisahkan dari sebuah huruf, karena berfungsi membedakan huruf yang berlainan bunyi.

Tabel 2.1 dan 2.2 menunjukkan pedoman transliterasi Arab Latin yang merupakan hasil keputusan bersama Menteri Agama dan Menteri Pendidikan dan Kebudayaan R.I. Nomor: 158 Tahun 1987 dan Nomor: 0543b/U/1987.

2.3.1 Konsonan

Tabel 2.1 menunjukkan daftar hurub bahasa Arab dan transliterasinya ke dalam huruf Latin.

Tabel 2.1 Daftar huruf bahasa Arab dan transliterasinya

Huruf Arab	Nama	Huruf Latin	Nama
ا	Alif	Tidak dilambangkan	Tidak dilambangkan
ب	Ba	B	Be
ت	Ta	T	Te
ث	Ša	Š	Es (dengan titik di atas)
ج	Jim	J	Je
ح	Ḥa	Ḥ	Ha (dengan titik di bawah)
خ	Kha	Kh	Ka dan Ha
د	Dal	D	De
ذ	Žal	Ž	Zet (dengan titik di atas)
ر	Ra	R	Er
ز	Zai	Z	Zet
س	Sin	S	Es
ش	Syin	Sy	Es dan Ye

Tabel 2.1 Daftar huruf bahasa Arab dan transliterasinya (lanjutan)

Huruf Arab	Nama	Huruf Latin	Nama
ص	Şad	Ş	Es (dengan titik di bawah)
ض	Ðad	Ð	De (dengan titik di bawah)
ط	Ṭa	Ṭ	Te (dengan titik di bawah)
ظ	Ẓa	Ẓ	Zet (dengan titik di bawah)
ع	‘Ain	‘ —	Apostrof terbalik
غ	Gain	G	Ge
ف	Fa	F	Ef
ق	Qof	Q	Qi
ك	Kaf	K	Ka
ل	Lam	L	El
م	Mim	M	Em
ن	Nun	N	En
و	Wau	W	We
ه	Ha	H	Ha
ء	Hamzah	— ’	Apostrof
ي	Ya	Y	Ye

Hamzah yang terletak di akhir atau di tengah ditulis dengan tanda (’), sedangkan hamzah yang terletak di awal kata mengikuti vokalnya diberi tanda apapun.

2.3.2 Vokal

Vokal bahasa Arab sama seperti vokal bahasa Indonesia, terdiri atas vokal tunggal dan vokal rangkap. Tabel 2.2 menunjukkan vokal tunggal bahasa Arab yang lambangnya berupa tanda atau *harakat*.

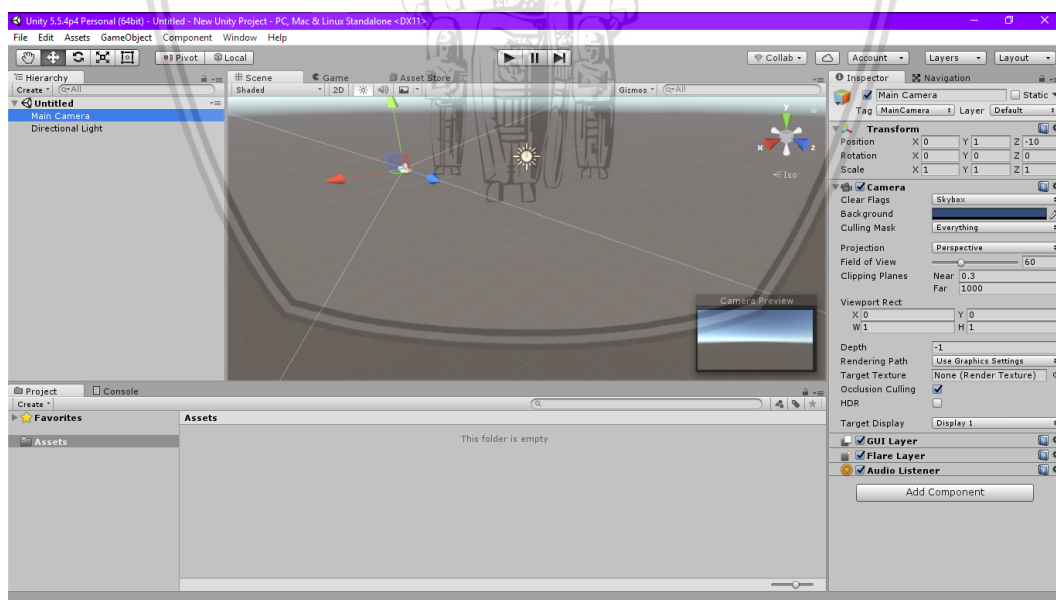
Tabel 2.2 Vokal tunggal bahasa Arab

Tanda	Nama	Huruf latin	Nama
اَ	<i>Fatḥah</i>	A	A
اِ	<i>Kasrah</i>	I	I
اُ	<i>Ḍammah</i>	U	U

2.4 Aplikasi Pendukung

2.4.1 Unity

Unity adalah sebuah *multi platform game engine* yang dikembangkan oleh Unity Technologies, berfokus pada pengembangan video game dan simulasi untuk komputer, konsol, dan perangkat *mobile*. Unity mendukung grafis 2D dan 3D, fungsi *drag and drop*, dan *scripting* menggunakan C#. Dua bahasa pemrograman lain yg bisa digunakan dalam unity adalah Boo dan UnityScript. Namun Boo tidak didukung lagi semenjak rilisnya Unity 5, sedangkan UnityScript tidak didukung lagi semenjak rilisnya Unity 2017.1 (Fine, 2017). Gambar 2.3 merupakan tampilan antarmuka dari aplikasi unity.



Gambar 2.3 Unity interface

Unity mendukung API (*Application Programming Interface*) grafis Direct3D di Windows dan Xbox One, OpenGL di Linux, macOS, dan Windows, OpenGL ES di Android dan iOS, WebGL di web, dan API yang ada pada konsol *video game*. Selain

itu unity juga mendukung *low-level* API Metal di iOS dan macOS, Vulkan di Android, Linux, dan Windows, serta Direct3D 12 di Windows dan Xbox One.

Dalam game 2D, unity dapat memasukkan *sprite* dan perender 2D tingkat lanjut. Sedangkan untuk game 3D, unity memungkinkan spesifikasi pengaturan kompresi tekstur dan resolusi untuk setiap platform yang didukung, dan memberikan dukungan untuk *reflection mapping*, *bump mapping*, *screen space ambient occlusion* (SSAO), *parallax mapping*, bayangan dinamis menggunakan *shadow maps*, *render-to-texture* dan efek *full-screen post-processing*.

Unity juga menawarkan layanan kepada para pengembang seperti Unity Ads, Unity Analytics, Unity Certification, Unity Cloud Build, Unity Everplay, Unity IAP, Unity Multiplayer, Unity Performance Reporting, dan Unity Collaborate.

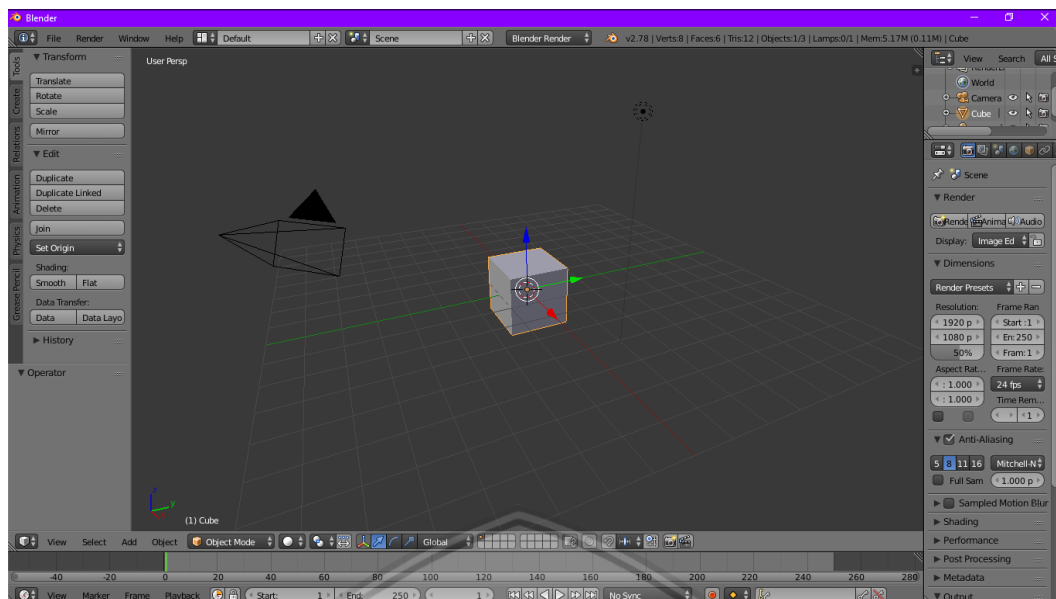
2.4.2 Vuforia SDK

Vuforia atau yang dulu dikenal dengan Qualcomm Company Augmented Reality (QCAR) adalah sebuah *Software Development Kit* (SDK) untuk perangkat *mobile* yang memungkinkan pembuatan aplikasi *augmented reality*. Vuforia pertama kali keluar untuk platform Android, dan kemudian diperluas sehingga dapat digunakan untuk perangkat iOS juga (Cushnan & Habbak, 2013).

Vuforia menggunakan teknologi *Computer Vision* untuk melacak dan mengenali gambar target dan objek 3D sederhana seperti kotak secara *real time*. Kemampuan registrasi gambar ini memungkinkan para pengembang untuk mengorientasikan dan memposisikan objek virtual, seperti model 3D dan media yang lainnya dengan gambar dari dunia nyata saat dilihat melalui kamera perangkat *mobile*. Vuforia juga memberikan sejumlah solusi *tracking* untuk beberapa situasi seperti *ARCamera*, *Image Target*, *Frame Marker*, *Multi-Targets*, dan *Virtual Button* (Cushnan & Habbak, 2013).

2.4.3 Blender 3D

Blender adalah sebuah perangkat lunak grafis komputer bersifat gratis dan *open source* yang digunakan untuk membuat animasi film, efek visual, seni, model cetak 3D, aplikasi interaktif 3D, dan video game. Blender dibuat oleh Ton Roosendaal pada tahun 1995 sebagai perangkat lunak untuk sebuah studio animasi Belanda yang bernama NeoGeo. Pada awalnya blender dijual untuk studio lain juga, namun pada awal tahun 2002 blender harus diberhentikan karena masalah penurunan ekonomi. Agar proyeknya tetap hidup, Roosendaal meyakinkan para investor untuk mengubah blender menjadi GNU General Public License dengan hanya mengumpulkan 100.000 Euro dalam waktu tujuh minggu. Sejak saat itu blender telah menjadi gratis dan terus berkembang berkat sumbangan dermawan dari komunitas (Lampel, 2015). Gambar 2.4 merupakan tampilan antarmuka aplikasi blender.

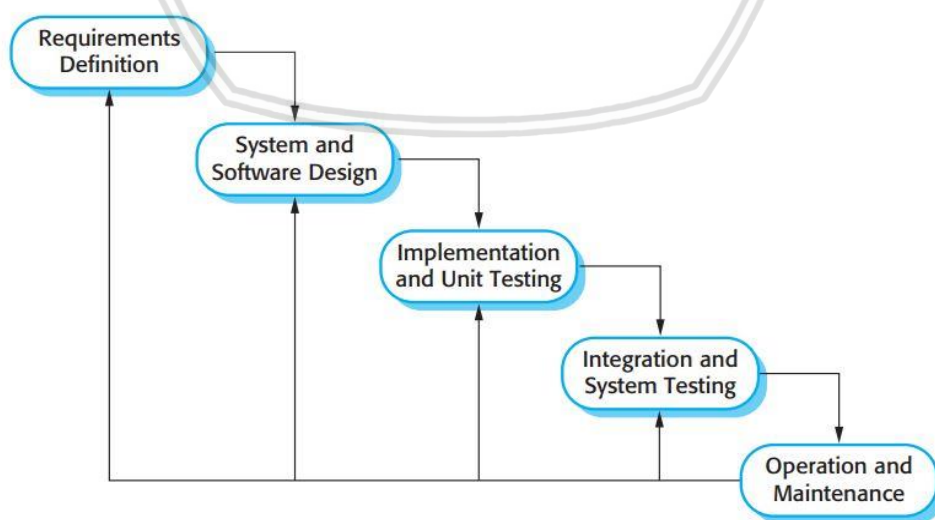


Gambar 2.4 Blender interface

Fitur-fitur blender meliputi pemodelan 3D, teksturing, *UV unwrapping*, editing grafis *raster*, *ringging* dan *skinning*, simulasi asap dan cairan, simulasi partikel, simulasi *soft body*, memahat, animasi, mencocokkan gerakan, *tracking* kamera, serta pengeditan dan pengomposisian video.

2.5 Metode *Waterfall*

Dalam pengembangan sistem, metode *waterfall* adalah metode dimana perubahan dari satu tahapan ke tahapan yang lain dilakukan secara berurutan. Dalam metode *waterfall* ini, sebuah tahapan akan diselesaikan terlebih dahulu sebelum melanjutkan ke tahapan yang berikutnya. Gambar 2.5 merupakan contoh dari tahapan metode *waterfall*.



Gambar 2.5 Contoh metode *waterfall*

Menurut Sommerville (2011) metode *waterfall* memiliki lima tahapan utama yang secara langsung mencerminkan kegiatan pengembangan, yaitu *requirements analysis and definition*, *system and software design*, *implementation and unit testing*, *integration and system testing*, dan *operation and maintenance*.

1. **Requirements Analysis and Definition**

Requirements analysis and definition merupakan tahapan yang menetapkan fitur, masalah, serta tujuan dari sistem melalui konsultasi dengan pengguna sistem. Semua hal tersebut akan ditetapkan secara rinci yang nantinya berfungsi sebagai spesifikasi sistem.

2. **System and Software Design**

System and software design merupakan tahapan yang akan membentuk suatu arsitektur sistem berdasarkan persyaratan yang sudah ditetapkan. Pada tahapan ini juga dilakukan penggambaran dan identifikasi terhadap abstraksi dasar sistem perangkat lunak beserta hubungannya.

3. **Implementation and Unit Testing**

Pada tahapan *implementation and unit testing*, perangkat lunak akan direalisasikan sebagai satu set program atau unit program. Pengujian unit melibatkan verifikasi bahwa setiap unit sudah memenuhi spesifikasinya.

4. **Integration and System Testing**

Pada tahap *integration and system testing*, setiap unit program terintegrasi satu sama lain dan diuji sebagai suatu sistem yang utuh untuk memastikan persyaratan dari sistem terpenuhi. Setelah pengujian, sistem akan dikirim ke pengguna sistem.

5. **Operation and Maintenance**

Pada tahap ini sistem diinstal dan mulai digunakan. Selain itu juga dilakukan perbaikan untuk kesalahan yang tidak ditemukan pada tahapan-tahapan sebelumnya dan dilakukan peningkatan layanan sistem.



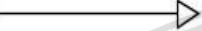


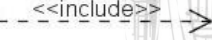
2.6 UML (*Unified Modeling Language*)

UML (*Unified Modeling Language*) adalah bahasa pemodelan untuk perangkat lunak atau sistem yang berparadigma berorientasi objek. Pemodelan sesungguhnya digunakan untuk penyederhanaan permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipahami dan dipelajari (Nugroho, 2010).

2.6.1 *Use case diagram*

Use case diagram merupakan representasi visual yang mewakili interaksi antara pengguna dengan sistem untuk menunjukkan hubungan antara pengguna dan bagaimana peran-peran menggunakan sistem. Tabel 2.3 merupakan simbol-simbol yang digunakan untuk membuat *use case diagram*.


Tabel 2.3 Deskripsi simbol-simbol *use case diagram*

Nama	Simbol	Deskripsi
Aktor		Aktor adalah entitas yang berinteraksi dengan suatu sistem, digunakan untuk mewakili pengguna sistem.
Use Case		Use case merepresentasikan tujuan pengguna yang dapat dicapai dengan mengakses sistem atau aplikasi perangkat lunak.
Generalisasi		Generalisasi digunakan untuk mewakili hubungan warisan antara model elemen dengan tipe yang sama.
Asosiasi		Asosiasi digunakan pada aktor dan use case untuk menunjukkan komunikasi antara aktor dan use case.
Extend		Extend merupakan relasi yang digunakan antar use case untuk menunjukkan bahwa suatu use case dapat berdiri sendiri tanpa use case lainnya.
Include		Include merupakan relasi yang digunakan antar use case untuk menunjukkan bahwa suatu use case memerlukan use case lainnya untuk menjalankan fungsinya.





2.6.2 Activity diagram

Activity diagram adalah diagram UML yang menggambarkan aspek dinamis dari sistem. *Activity diagram* pada dasarnya adalah *flowchart* untuk merepresentasikan aliran dari satu aktivitas ke aktivitas lainnya. Aktivitas dapat digambarkan sebagai operasi dari sistem. Aliran kontrol diambil dari satu sistem operasi ke operasi lainnya. Aliran ini dapat berurutan, bercabang, atau bersamaan. Tabel 2.4 merupakan simbol-simbol yang digunakan untuk membuat *activity diagram*.

Tabel 2.4 Deskripsi simbol-simbol *activity diagram*

Nama	Simbol	Deskripsi
Start Point		Start Point/Initial State merupakan status awal aktivitas sistem.

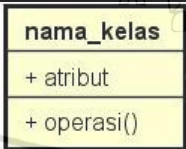



Tabel 2.4 Deskripsi simbol-simbol *activity diagram* (lanjutan)

Nama	Simbol	Deskripsi
<i>Activity</i>		<i>Activity</i> menggambarkan aktivitas yang dilakukan oleh sistem.
<i>Action Flow</i>		<i>Action flow</i> menggambarkan transisi dari satu aksi ke aksi lainnya.
<i>Decision</i>		<i>Decision</i> menggambarkan keputusan dengan jalur alternatif ketika suatu aktivitas membutuhkan keputusan sebelum pindah ke aktivitas berikutnya.
<i>End Point</i>		<i>End Point</i> menggambarkan status akhir yang dilakukan sistem.

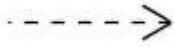

2.6.3 Class diagram

Class diagram adalah jenis diagram yang menggambarkan struktur sistem dengan menunjukkan kelas, atribut, operasi, serta hubungan antara objek lainnya dari sistem. Tabel 2.5 merupakan simbol-simbol yang digunakan untuk membuat *class diagram*.

Tabel 2.5 Deskripsi simbol-simbol *class diagram*

Nama	Simbol	Deskripsi
Kelas		Kelas merepresentasikan entitas dengan karakteristik umum.
<i>Association</i>		<i>Association</i> menggambarkan relasi antar kelas dengan mana umum.
<i>Association Navigability</i>		<i>Association Navigability</i> menggambarkan relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain.
Generalisasi		Generalisasi menggambarkan relasi antara dua kelas dimana satu kelas adalah khusus dari yang lain.

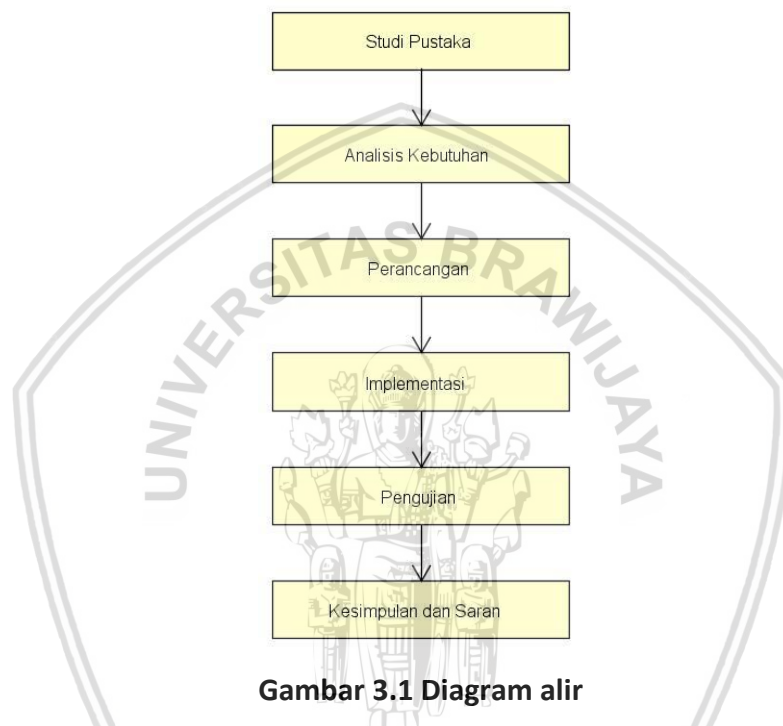
Tabel 2.5 Deskripsi simbol-simbol *class diagram* (lanjutan)

Nama	Simbol	Deskripsi
<i>Dependency</i>		<i>Dependency</i> menggambarkan relasi antar kelas dengan makna ketergantungan antar kelas.
<i>Aggregation</i>		<i>Aggregation</i> menggambarkan relasi antar kelas dengan makna smua-bagian (<i>whole-part</i>).



BAB 3 METODOLOGI PENELITIAN

Bab ini menjelaskan tentang langkah-langkah yang akan dilakukan dalam melakukan penelitian mulai dari studi pustaka, analisis kebutuhan, perancangan, implementasi, pengujian, sampai penarikan kesimpulan dan saran. Metode yang akan digunakan dalam pengembangan aplikasi ini adalah metode *waterfall* dengan menggunakan pendekatan berorientasi objek dan hanya sampai pada tahapan pengujian sistem. Gambar 3.1 menunjukkan diagram alir dari penelitian secara umum.



Gambar 3.1 Diagram alir

3.1 Studi Pustaka

Studi pustaka menjelaskan tentang dasar-dasar teori yang digunakan untuk menunjang perancangan aplikasi *augmented reality* media belajar pengenalan huruf hijaiyah. Teori-teori pendukung tersebut meliputi jurnal, buku referensi, skripsi yang sudah ada, serta kumpulan artikel dan dokumentasi dari internet. Berikut adalah teori pendukungnya:

1. Media Pembelajaran
2. Huruf Hijaiyah
3. *Augmented Reality*
4. Unity, Vuforia SDK, dan Blender 3D
5. Metode *waterfall*
6. UML (*Unified Modeling Language*)

3.2 Analisis Kebutuhan (*Requirement Analysis and Definition*)

Dalam membangun sebuah aplikasi, analisis kebutuhan digunakan untuk mengetahui apa saja kebutuhan dari sistem. Adapun kebutuhan dari aplikasi *augmented reality* sebagai media belajar pengenalan huruf hijaiyah pada *platform* android dapat dibagi menjadi beberapa bagian, diantaranya:

1. Kebutuhan fungsionalitas
2. Kebutuhan non fungsionalitas

3.3 Perancangan (*System and Software Design*)

Tahap perancangan aplikasi dilakukan setelah semua kebutuhan aplikasi telah didapatkan dari analisis kebutuhan. Pendekatan yang digunakan dalam tahap perancangan aplikasi adalah pendekatan berorientasi objek dan menggunakan UML untuk menggambarkan bagaimana aplikasi akan dibuat. Perancangan *marker* dilakukan untuk membuat gambar yang akan digunakan sebagai *marker* lebih mudah dikenali sistem dan untuk menghindari terjadinya kekeliruan pada saat aplikasi mendeteksi *marker*, sedangkan perancangan antarmuka aplikasi dilakukan untuk menentukan bagaimana *user* dapat berinteraksi dengan sistem.

3.4 Implementasi (*Implementation and Unit Testing*)

Tahap implementasi dilakukan setelah setelah semua perancangan sudah selesai dilakukan. Pada tahap ini semua hasil dari analisis dan perancangan akan di implementasikan kedalam sebuah aplikasi. Implementasi aplikasi dimulai dengan penjabaran spesifikasi lingkungan perancangan aplikasi, implementasi *marker*, implementasi konten 3D, implementasi antarmuka aplikasi, sampai tahap implementasi kode program sesuai dengan perancangan yang telah dilakukan sebelumnya.

3.5 Pengujian (*Integration and System Testing*)

Pengujian atau evaluasi aplikasi dilakukan untuk menunjukkan bahwa aplikasi yang dibuat dapat bekerja dan berjalan sesuai dengan kebutuhan. Pengujian juga dilakukan untuk menemukan kesalahan ataupun kekurangan dari aplikasi yang telah dibuat. Dalam penelitian ini, pengujian yang dilakukan adalah pengujian fungsional, pengujian aplikasi, dan kondisi optimal *marker* berdasarkan besar dan jarak. Pengujian dilakukan dengan menggunakan metode *black box testing*.

Pengujian aplikasi dilakukan dengan melihat peningkatan nilai membaca siswa. Tahapannya adalah siswa diajarkan huruf hijaiyah secara tradisional lalu dilakukan pengambilan nilai membacanya. Kemudian siswa menggunakan aplikasi dan diambil nilai membacanya lagi untuk melihat seberapa besar peningkatan nilai membaca siswa. Pengujian dilakukan pada siswa kelas I SD Khoiru Ummah Kranggan kota Bekasi dengan bantuan dari guru yang mengajar untuk menggunakan aplikasi. Pengujian dilakukan pada tanggal 15 Februari 2018.

Proses analisis dilakukan setelah semua tahap sebelumnya sudah selesai. Proses ini bertujuan untuk mengetahui hasil dari pengujian aplikasi yang nantinya dapat ditarik sebagai kesimpulan.

3.6 Penarikan Kesimpulan dan Saran

Penarikan kesimpulan diambil berdasarkan hasil pengujian dan analisis yang telah dilakukan terhadap aplikasi yang dibangun dengan tujuan untuk mengetahui apakah permasalahan yang telah dirumuskan dapat diselesaikan atau tidak. Pada bagian akhir dari penelitian terdapat saran yang bertujuan untuk memperbaiki kesalahan, kekurangan, dan menyempurnakan penulisan.



BAB 4 PERANCANGAN

Pada bab ini perancangan dibagi menjadi tiga tahap yaitu perancangan umum sistem, analisis kebutuhan perangkat lunak, dan perancangan perangkat lunak.

4.1 Perancangan Umum Sistem

Perancangan umum sistem adalah tahapan awal dari perancangan perangkat lunak yaitu dengan mendefinisikan kebutuhan-kebutuhan fungsional sebagai gambaran sistem yang akan dibentuk. Perancangan dilakukan untuk merepresentasikan sistem secara umum.

Sistem yang dirancang akan berjalan pada perangkat *mobile* dengan sistem operasi Android sebagai alat input serta output dari sistem. Operasi sistem yang digunakan minimal Android versi 4.0. Pada aplikasi ini kamera tetap menjadi sebuah alat yang digunakan untuk menangkap gambar yang nantinya akan diproses oleh sistem yang kemudian akan menampilkan objek 3D.

Dalam pembuatan objek 3D digunakan perangkat lunak Blender yang di karenakan perangkat lunak ini bersifat gratis dan *open source*, tampilannya yang bisa diubah-ubah sesuka hati, serta karena project kerja perangkat lunak ini bisa dikerjakan di hampir semua perangkat lunak 3D komersial lainnya. Sedangkan untuk menampilkan objek 3D digunakan target gambar atau *marker* dengan bantuan vuforia. Selain mendukung berbagai jenis *marker* 2D, vuforia juga mendukung *marker* 3D seperti objek berbentuk kubik, silinder, ataupun objek 3D lainnya.

4.2 Analisis Kebutuhan Perangkat Lunak

Proses analisis kebutuhan perangkat lunak dilakukan untuk menggambarkan kebutuhan-kebutuhan yang harus disediakan oleh aplikasi untuk memenuhi kebutuhan dari pengguna. Pada proses analisis ini terdapat beberapa bagian, yaitu indentifikasi aktor, penjabaran fungsional dengan menggunakan *use case diagram*, serta kebutuhan non fungsional.

4.2.1 Identifikasi aktor

Tahap ini dilakukan untuk mengidentifikasi siapa saja yang nantinya akan berinteraksi dengan aplikasi.

Tabel 4.1 Identifikasi aktor

Aktor	Deskripsi
User / Pengguna	Pengguna adalah orang yang menggunakan aplikasi. Pengguna akan dapat melihat objek 3D, pengucapan huruf, transliterasi, serta animasi penulisan dari huruf.

4.2.2 Analisis kebutuhan fungsional

Kebutuhan fungsional pada aplikasi ini akan diuraikan dan dispesifikasikan pada Tabel 4.2 dengan penomoran menggunakan SRS (*Software Requirement Specification*).

Tabel 4.2 Spesifikasi kebutuhan fungsional *user*

Nomor SRS	Kebutuhan	<i>Use Case</i>
SRS_000_01	Aplikasi harus mampu menampilkan panduan penggunaan aplikasi dan penjelasan tombol-tombol.	Melihat halaman bantuan.
SRS_000_02	Aplikasi harus mampu menampilkan informasi tentang aplikasi.	Melihat halaman tentang.
SRS_000_03	Aplikasi harus mampu menutup aplikasi	Keluar dari aplikasi
SRS_001_01	Aplikasi harus mampu mengidentifikasi dan menampilkan objek 3D dari setiap <i>marker</i> yang berbeda.	Memulai mode AR.
SRS_002_01	Aplikasi harus mampu menampilkan varian dari setiap huruf.	Lihat varian huruf.
SRS_002_02	Aplikasi harus mampu menampilkan animasi penulisan setiap huruf.	Lihat animasi penulisan.
SRS_003_01	Aplikasi harus mampu menampilkan transliterasi dari setiap huruf.	Lihat transliterasi.
SRS_003_02	Aplikasi harus mampu mengeluarkan suara pengucapan dari setiap huruf.	Dengar pembacaan huruf.

4.2.3 Analisis kebutuhan non fungsional

Analisis kebutuhan non fungsional bertujuan untuk membuat sistem agar dapat berjalan dengan baik dengan cara mengetahui spesifikasi yang dibutuhkan oleh sistem. Kebutuhan non fungsional tersebut adalah *compatibility* dan *usability*.

Tabel 4.3 Spesifikasi kebutuhan non fungsional

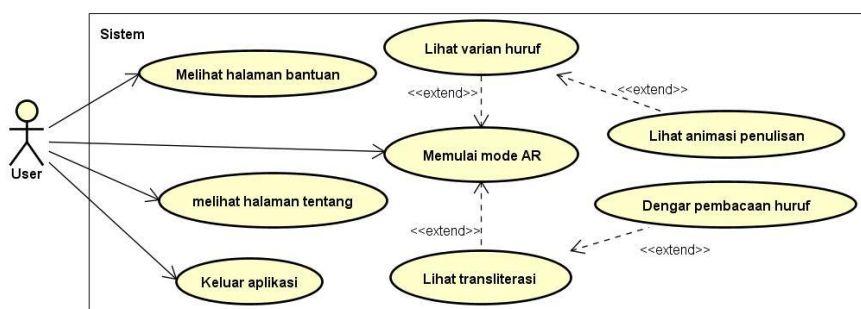
Parameter	Deskripsi kebutuhan
<i>Compatibility</i>	Aplikasi harus dapat berjalan pada perangkat <i>mobile</i> dengan sistem operasi Android minimal versi 4.0 dan dalam keadaan <i>offline</i> .
<i>Usability</i>	Tampilan dan desain antarmuka dirancang sesederhana mungkin dan menggunakan bahasa Indonesia sehingga memudahkan <i>user</i> dalam menggunakan dan membaca informasi yang ada pada aplikasi.

4.3 Perancangan Perangkat Lunak

Tahap perancangan perangkat lunak dalam penelitian ini menggunakan pendekatan berorientasi objek, yaitu memandang sistem yang akan dibangun sebagai kumpulan-kumpulan objek yang berkomunikasi dengan objek-objek dari dunia nyata. *Unified Modeling Language* (UML) digunakan untuk menggambarkan perancangannya.

4.3.1 Use case diagram

Diagram *use case* merupakan salah satu diagram yang digunakan untuk menunjukkan aksi-aksi yang dilakukan dari aktor ke sistem serta menggambarkan kebutuhan-kebutuhan fungsionalitas dari sistem. Gambar 4.1 menunjukkan diagram *use case* pada aplikasi dari sisi *user*.



Gambar 4.1 Use case diagram

Dalam aplikasi ini *user* dapat melihat objek huruf hijaiyah, melihat varian dari huruf hijaiyah, melihat animasi penulisan huruf hijaiyah, melihat transliterasi huruf hijaiyah, serta mendengar pengucapannya.

Senario *use case* akan digunakan untuk menjabarkan diagram *use case* dari *user* yang ada pada Gambar 4.1.

4.3.1.1 Skenario *use case* melihat halaman bantuan

Tabel 4.4 Skenario *use case* melihat halaman bantuan

Nama	Melihat halaman bantuan
Kode SRS	SRS_000_01
Tujuan	Untuk melihat halaman bantuan.
Deskripsi (Brief Description)	<i>Use case</i> ini memungkinkan <i>user</i> untuk melihat halaman bantuan yang berisi panduan penggunaan aplikasi dan penjelasan tombol-tombol.
Aktor	<i>User</i> / Pengguna
Kondisi awal (Pre-Condition)	Aktor harus terlebih dahulu menjalankan aplikasi sebelum <i>use case</i> dimulai dan masuk pada halaman utama.
Flow of Events	
Alur Utama (Basic Flow) <ol style="list-style-type: none"> 1. Sistem meminta aktor memilih menu “Bantuan” yang ada pada halaman utama. 2. Sistem masuk ke dalam halaman bantuan. 3. Sistem akan menampilkan panduan penggunaan aplikasi dan penjelasan tombol-tombol. 	
Kondisi Akhir (Post-Condition)	Sistem menampilkan panduan penggunaan aplikasi dan penjelasan tombol-tombol.

4.3.1.2 Skenario *use case* melihat halaman tentang

Tabel 4.5 Skenario *use case* melihat halaman tentang

Nama	Melihat halaman tentang
Kode SRS	SRS_000_02

Tabel 4.5 Skenario use case melihat halaman tentang (lanjutan)

Tujuan	Untuk melihat halaman tentang.
Deskripsi (Brief Description)	<i>Use case</i> ini memungkinkan <i>user</i> untuk melihat halaman tentang yang berisi informasi tentang aplikasi.
Aktor	<i>User</i> / Pengguna
Kondisi awal (Pre-Condition)	Aktor harus terlebih dahulu menjalankan aplikasi sebelum <i>use case</i> dimulai dan masuk pada halaman utama.
Flow of Events	
Alur Utama (Basic Flow) <ol style="list-style-type: none"> 1. Sistem meminta aktor memilih menu “Tentang” yang ada pada halaman utama. 2. Sistem masuk ke dalam halaman tentang. 3. Sistem akan menampilkan informasi tentang aplikasi. 	
Kondisi Akhir (Post-Condition)	Sistem menampilkan informasi tentang aplikasi.

4.3.1.3 Skenario use case keluar aplikasi

Tabel 4.6 Skenario use case keluar aplikasi

Nama	Keluar aplikasi
Kode SRS	SRS_000_03
Tujuan	Untuk keluar dari aplikasi.
Deskripsi (Brief Description)	<i>Use case</i> ini memungkinkan <i>user</i> untuk keluar dari aplikasi.
Aktor	<i>User</i> / Pengguna
Kondisi awal (Pre-Condition)	Aktor harus terlebih dahulu menjalankan aplikasi sebelum <i>use case</i> dimulai dan masuk pada halaman utama.
Flow of Events	

Tabel 4.6 Skenario use case keluar aplikasi (lanjutan)

Alur Utama (Basic Flow) <ol style="list-style-type: none"> 1. Sistem meminta aktor memilih menu “Keluar” yang ada pada halaman utama. 2. Sistem akan menampilkan konfirmasi untuk keluar. 3. Sistem meminta aktor memilih “Ya”. 4. Sistem akan menutup aplikasi. 	
Kondisi Akhir (Post-Condition)	Apabila aktor memilih “Ya” pada saat konfirmasi, sistem akan menutup aplikasi. Apabila aktor memilih “Tidak” pada saat konfirmasi, sistem akan kembali ke halaman utama.

4.3.1.4 Skenario use case memulai mode AR

Tabel 4.7 Skenario use case memulai mode AR

Nama	Memulai mode AR
Kode SRS	SRS_001_01
Tujuan	Untuk memasuki mode AR dan mengetahui valid atau tidaknya <i>marker</i> .
Deskripsi (Brief Description)	<i>Use case</i> ini memungkinkan <i>user</i> untuk masuk ke dalam mode AR dan mengetahui valid atau tidaknya <i>marker</i> . Apabila <i>marker</i> valid, maka fitur-fitur akan aktif.
Aktor	User / Pengguna
Kondisi awal (Pre-Condition)	Aktor harus terlebih dahulu menjalankan aplikasi dan masuk pada halaman utama.
Flow of Events	
Alur Utama (Basic Flow) <ol style="list-style-type: none"> 1. Sistem meminta aktor memilih menu “Mulai” yang ada pada halaman utama. 2. Sistem masuk ke dalam <i>camera view</i>. 3. Sistem meminta aktor mengarahkan kamera pada <i>marker</i>. 4. Sistem akan mengidentifikasi valid atau tidaknya <i>marker</i>. 	

Tabel 4.7 Skenario *use case* memulai mode AR (lanjutan)

Kondisi Akhir (<i>Post-Condition</i>)	Apabila <i>marker</i> valid, objek 3D serta fitur-fitur lainnya akan ditampilkan. Apabila tidak valid, maka sistem tidak akan menampilkan apa-apa.
---	--

4.3.1.5 Skenario *use case* lihat varian hurufTabel 4.8 Skenario *use case* lihat varian huruf

Nama	Lihat varian huruf
Kode SRS	SRS_002_01
Tujuan	Menampilkan varian huruf hijaiyah.
Deskripsi (<i>Brief Description</i>)	<i>Use case</i> ini memungkinkan <i>user</i> untuk melihat varian huruf hijaiyah yang ditampilkan.
Aktor	User / Pengguna
Kondisi awal (<i>Pre-Condition</i>)	<i>Marker</i> telah teridentifikasi oleh sistem.
Flow of Events	
Alur Utama (<i>Basic Flow</i>)	
<ol style="list-style-type: none"> 1. Sistem meminta aktor memilih fitur lihat varian yang ada setelah <i>marker</i> teridentifikasi. 2. Sistem akan menampilkan salah satu varian huruf hijaiyah dan tombol untuk melihat varian huruf hijaiyah lainnya sesuai dengan <i>marker</i> yang teridentifikasi. 3. Ketika aktor selesai melihat varian huruf hijaiyah, maka <i>use case</i> berakhir. 	
Kondisi Akhir (<i>Post-Condition</i>)	Sistem menampilkan salah satu varian huruf hijaiyah, tombol untuk melihat varian huruf hijaiyah lainnya dan tombol untuk melihat animasi penulisan sesuai dengan <i>marker</i> yang teridentifikasi.

4.3.1.6 Skenario *use case* lihat animasi penulisanTabel 4.9 Skenario *use case* lihat animasi penulisan

Nama	Lihat animasi penulisan
Kode SRS	SRS_002_02
Tujuan	Menampilkan animasi penulisan huruf hijaiyah.
Deskripsi (Brief Description)	<i>Use case</i> ini memungkinkan <i>user</i> untuk melihat animasi penulisan huruf hijaiyah yang ditampilkan.
Aktor	<i>User</i> / Pengguna
Kondisi awal (Pre-Condition)	Aktor telah memilih fitur lihat varian huruf yang ada setelah <i>marker</i> teridentifikasi.
Flow of Events	
Alur Utama (Basic Flow)	
<ol style="list-style-type: none"> 1. Sistem meminta aktor memilih fitur melihat animasi yang ada setelah memilih fitur lihat varian. 2. Sistem akan menampilkan animasi penulisan huruf hijaiyah sesuai dengan <i>marker</i> yang teridentifikasi. 3. Ketika aktor selesai melihat animasi penulisan huruf hijaiyah, maka <i>use case</i> berakhir. 	
Kondisi Akhir (Post-Condition)	Sistem menampilkan animasi penulisan huruf hijaiyah sesuai dengan <i>marker</i> yang teridentifikasi.

4.3.1.7 Skenario *use case* lihat transliterasiTabel 4.10 Skenario *use case* lihat transliterasi

Nama	Lihat transliterasi
Kode SRS	SRS_003_01
Tujuan	Menampilkan transliterasi dari huruf hijaiyah.
Deskripsi (Brief Description)	<i>Use case</i> ini memungkinkan <i>user</i> untuk melihat transliterasi dari huruf hijaiyah yang ditampilkan.
Aktor	<i>User</i> / Pengguna

Tabel 4.10 Skenario *use case* lihat transliterasi (lanjutan)

Kondisi awal (Pre-Condition)	<i>Marker</i> telah teridentifikasi oleh sistem.
Flow of Events	
Alur Utama (Basic Flow) <ol style="list-style-type: none"> 1. Sistem meminta aktor memilih fitur lihat transliterasi yang ada setelah <i>marker</i> teridentifikasi. 2. Sistem akan menampilkan transliterasi huruf hijaiyah sesuai dengan <i>marker</i> yang teridentifikasi. 3. Ketika aktor selesai melihat transliterasi huruf hijaiyah, maka <i>use case</i> berakhir. 	
Kondisi Akhir (Post-Condition)	Sistem menampilkan transliterasi huruf hijaiyah sesuai dengan <i>marker</i> yang teridentifikasi.

4.3.1.8 Skenario *use case* dengar pembacaan hurufTabel 4.11 Skenario *use case* dengar pembacaan huruf

Nama	Dengar pembacaan huruf
Kode SRS	SRS_003_02
Tujuan	Memutar suara pengucapan dari huruf hijaiyah.
Deskripsi (Brief Description)	<i>Use case</i> ini memungkinkan <i>user</i> untuk mendengar suara pembacaan dari huruf hijaiyah yang ditampilkan.
Aktor	<i>User</i> / Pengguna
Kondisi awal (Pre-Condition)	Aktor telah memilih fitur lihat transliterasi yang ada setelah <i>marker</i> teridentifikasi.
Flow of Events	
Alur Utama (Basic Flow) <ol style="list-style-type: none"> 1. Sistem meminta aktor memilih fitur dengar pembacaan huruf yang ada setelah memilih fitur lihat transliterasi. 2. Sistem akan memutar suara pembacaan huruf hijaiyah sesuai dengan <i>marker</i> yang teridentifikasi. 	

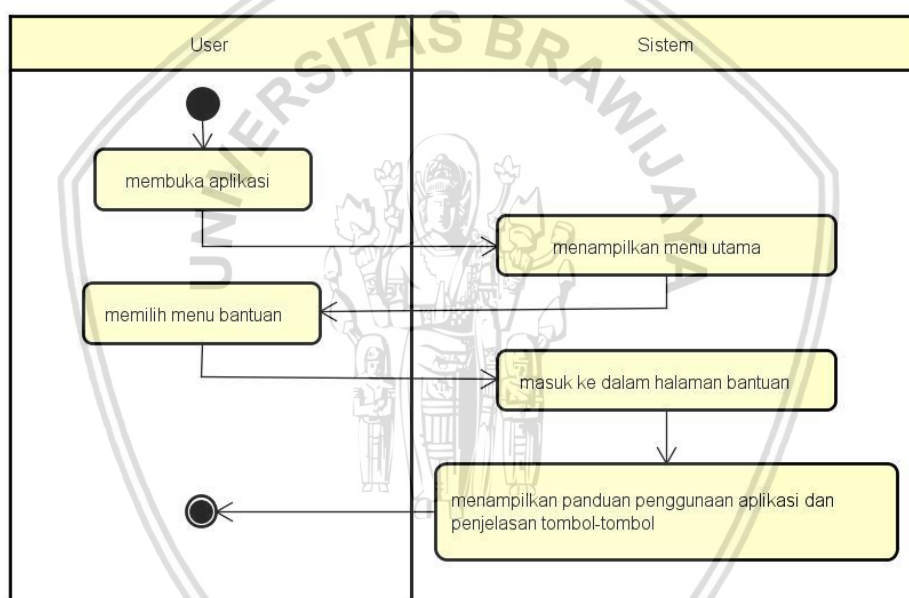
Tabel 4.11 Skenario *use case* dengar pembacaan huruf (lanjutan)

3. Ketika aktor selesai melihat transliterasi huruf hijiyah, maka <i>use case</i> berakhir.	
Kondisi Akhir (Post-Condition)	Sistem memutar suara pembacaan huruf hijaiyah sesuai dengan <i>marker</i> yang teridentifikasi.

4.3.2 Activity diagram

Setelah identifikasi aktor, selanjutnya adalah memodelkan secara grafis langkah-langkah dari setiap aktivitas yang dilakukan ke dalam sebuah *activity diagram*.

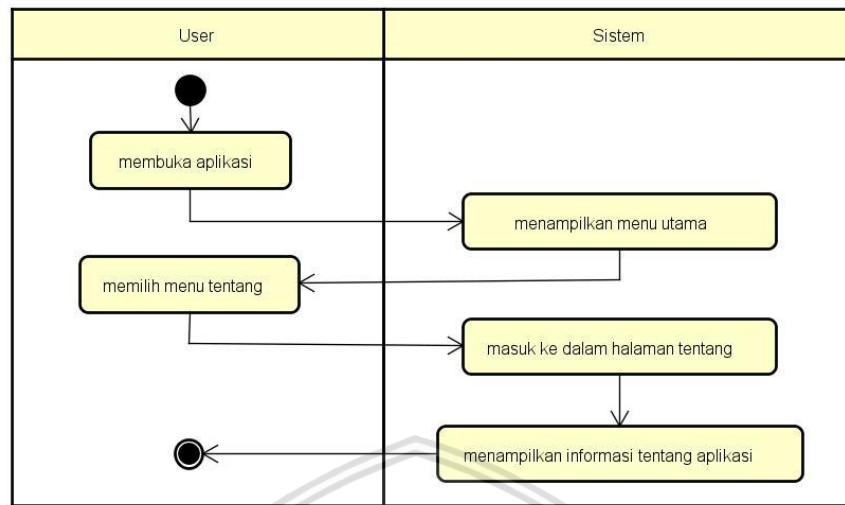
4.3.2.1 Activity diagram melihat halaman bantuan



Gambar 4.2 Activity diagram melihat halaman bantuan

Gambar 4.2 menggambarkan aktivitas yang dilakukan oleh *user* dan sistem. *User* membuka aplikasi, kemudian sistem akan menampilkan menu utama. Pada menu utama *user* memilih menu “bantuan”, kemudian sistem akan masuk pada halaman bantuan, lalu sistem akan menampilkan panduan penggunaan aplikasi dan penjelasan tombol-tombol.

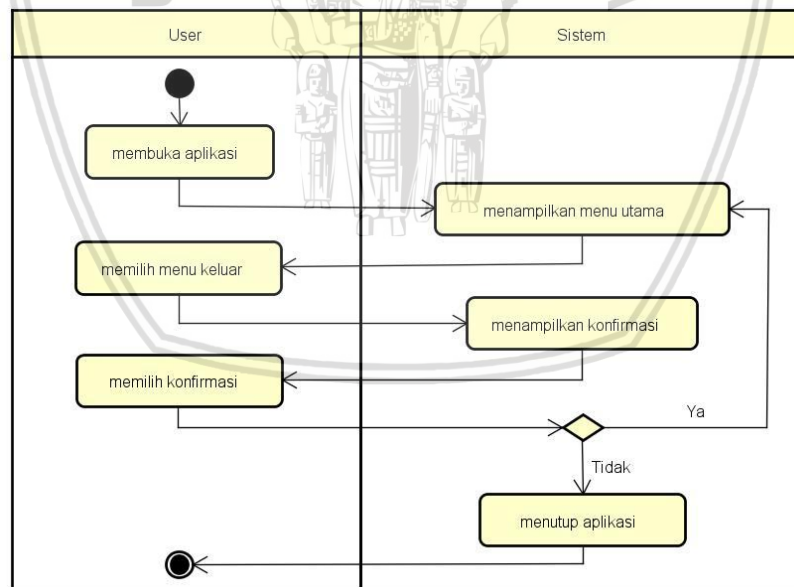
4.3.2.2 Activity diagram melihat halaman tentang



Gambar 4.3 Activity diagram melihat halaman tentang

Gambar 4.3 menggambarkan aktivitas yang dilakukan oleh *user* dan sistem. *User* membuka aplikasi, kemudian sistem akan menampilkan menu utama. Pada menu utama *user* memilih menu “tentang”, kemudian sistem akan masuk pada halaman tentang, lalu sistem akan menampilkan informasi tentang aplikasi.

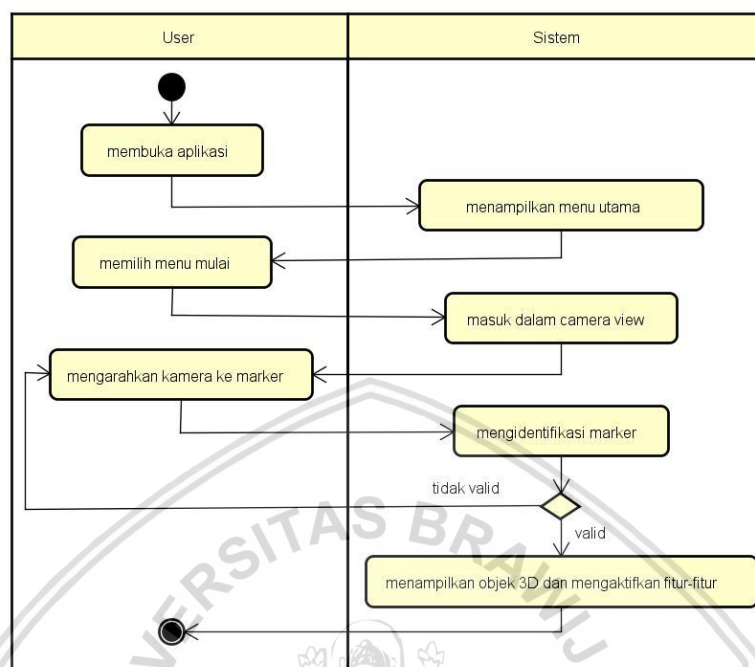
4.3.2.3 Activity diagram keluar aplikasi



Gambar 4.4 Activity diagram keluar aplikasi

Gambar 4.4 menggambarkan aktivitas yang dilakukan oleh *user* dan sistem. *User* membuka aplikasi, kemudian sistem akan menampilkan menu utama. Pada menu utama *user* memilih menu “keluar”, kemudian sistem akan menampilkan konfirmasi. Apabila *user* memilih “Ya”, sistem akan menutup aplikasi. Apabila *user* memilih “Tidak”, sistem akan kembali ke halaman utama.

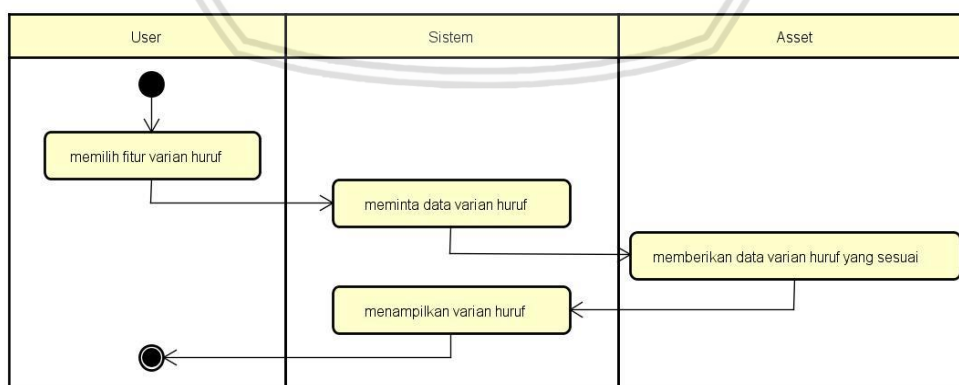
4.3.2.4 Activity diagram memulai mode AR



Gambar 4.5 Activity diagram memulia mode AR

Gambar 4.5 menggambarkan aktivitas yang dilakukan oleh *user* dan sistem. *User* membuka aplikasi, kemudian sistem akan menampilkan menu utama. Pada menu utama *user* memilih menu “mulai”, kemudian sistem akan masuk pada *camera mode*, lalu *user* mengarahkan kamera ke *marker* dan sistem akan mengidentifikasi *marker*. Jika *marker* valid, maka objek 3D akan muncul dan fitur akan aktif, jika tidak maka *user* mengarahkan kamera ke *marker* yang lain.

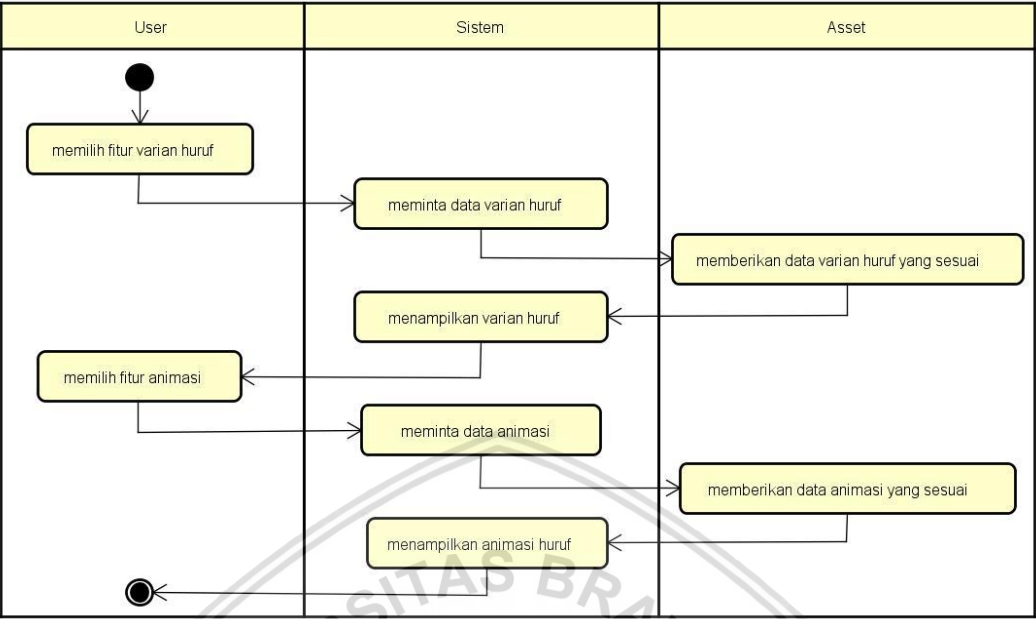
4.3.2.5 Activity diagram lihat varian huruf



Gambar 4.6 Activity diagram lihat varian huruf

Gambar 4.6 menggambarkan aktivitas yang dilakukan oleh *user* dan sistem. *User* akan memilih fitur varian huruf untuk melihat varian huruf, kemudian sistem akan menampilkan varian huruf sesuai dengan *marker* yang terdeteksi.

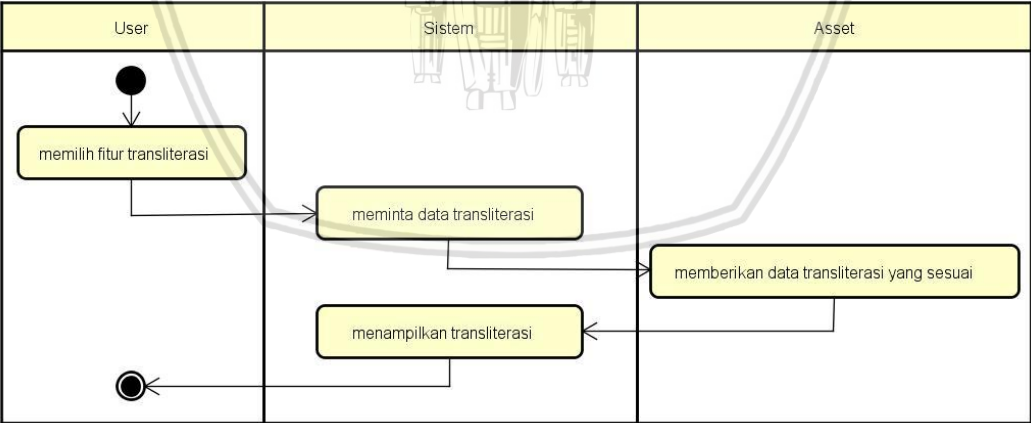
4.3.2.6 Activity diagram lihat animasi penulisan



Gambar 4.7 Activity diagram lihat animasi penulisan

Gambar 4.7 menggambarkan aktivitas yang dilakukan oleh *user* dan sistem. *User* akan memilih fitur varian untuk melihat varian huruf, kemudian sistem akan menampilkan varian huruf sesuai dengan *marker* yang terdeteksi, setelah itu *user* memilih fitur animasi, lalu sistem akan menampilkan animasi penulisan huruf.

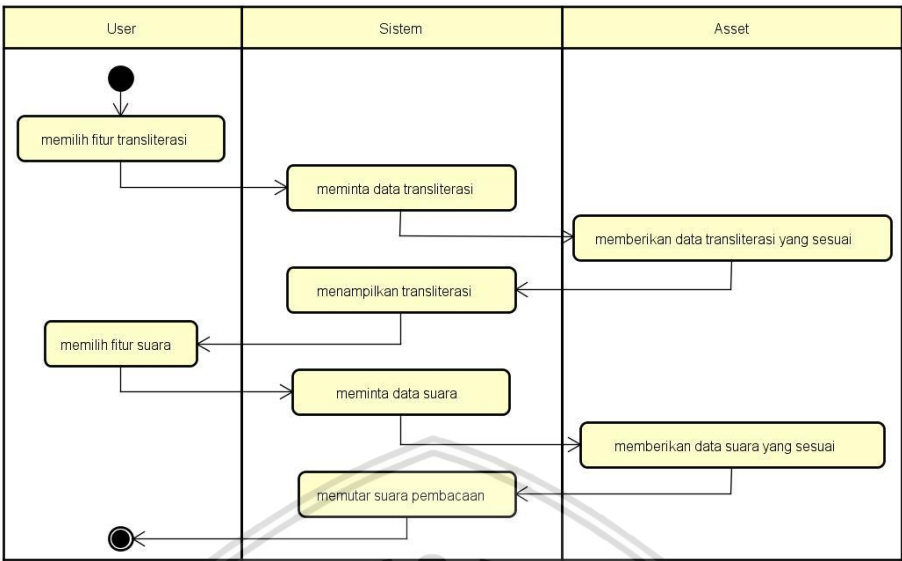
4.3.2.7 Activity diagram lihat transliterasi



Gambar 4.8 Activity diagram lihat transliterasi

Gambar 4.8 menggambarkan aktivitas yang dilakukan oleh *user* dan sistem. *User* memilih fitur transliterasi untuk melihat transliterasi huruf, kemudian sistem akan menampilkan transliterasi huruf sesuai dengan *marker* yang terdeteksi.

4.3.2.8 Activity diagram dengan pembacaan huruf

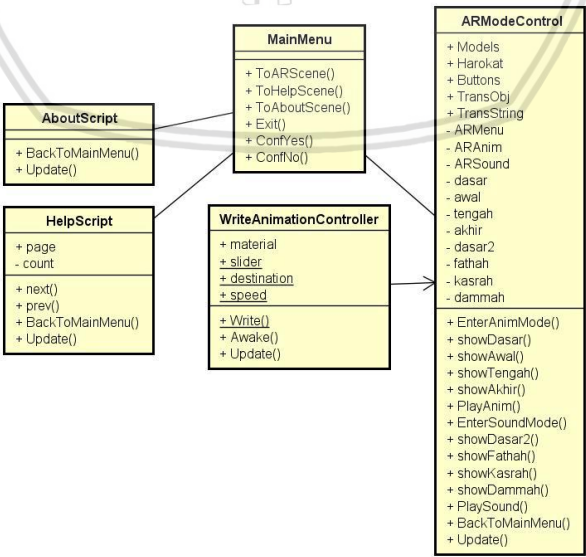


Gambar 4.9 Activity diagram dengan pembacaan huruf

Gambar 4.9 menggambarkan aktivitas yang dilakukan oleh *user* dan sistem. *User* akan memilih fitur transliterasi untuk melihat transliterasi huruf, kemudian sistem akan menampilkan transliterasi sesuai dengan *marker* yang terdeteksi, setelah itu *user* memilih fitur suara, lalu sistem akan memutar suara pembacaan huruf.

4.3.3 Class diagram

Class diagram merepresentasikan permodelan elemen-elemen *class* yang membentuk sebuah aplikasi. *Class* didapatkan dari *use case* yang dimodelkan dengan menganalisisnya secara detail.



Gambar 4.10 Class diagram

Class diagram yang ada pada Gambar 4.10 menunjukkan beberapa *class* utama yang menyusun aplikasi. Deskripsi *class* tersebut dijelaskan pada Tabel 4.12.

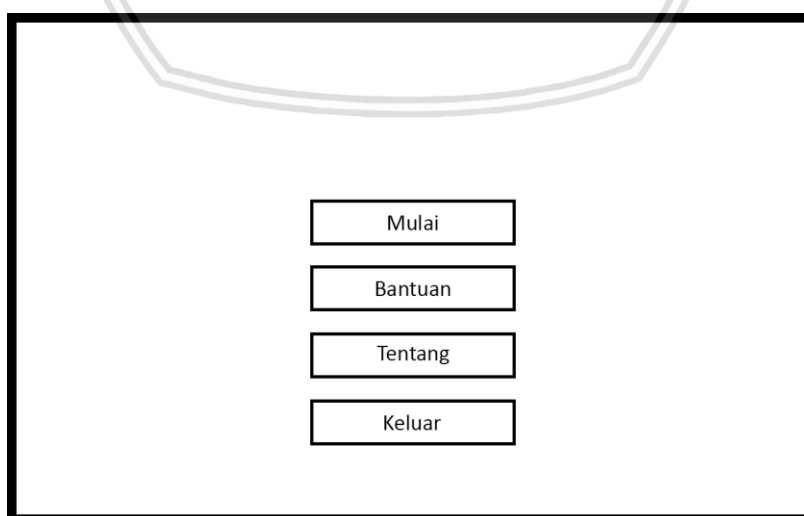
Tabel 4.12 Deskripsi *class diagram*

Nama Class	Deskripsi
MainMenu	<i>Class</i> ini mengatasi tampilan halaman pertama pada saat aplikasi dibuka, yaitu tampilan menu utama aplikasi.
HelpScript	<i>Class</i> ini mengatasi tampilan halaman yang berisi panduan penggunaan aplikasi dan penjelasan tombol-tombol.
AboutScript	<i>Class</i> ini mengatasi tampilan halaman yang berisi informasi tentang aplikasi.
ARModeControl	<i>Class</i> ini mengatasi halaman untuk mengidentifikasi <i>marker</i> , menampilkan fitur ketika <i>marker</i> valid, serta pemutaran suara pengucapan huruf.
WriteAnimationController	<i>Class</i> ini menangani pemutaran animasi pada objek 3D.

4.3.4 Perancangan antarmuka

Berikut ini adalah rancangan antarmuka dari aplikasi yang akan dibuat:

a. Tampilan menu utama



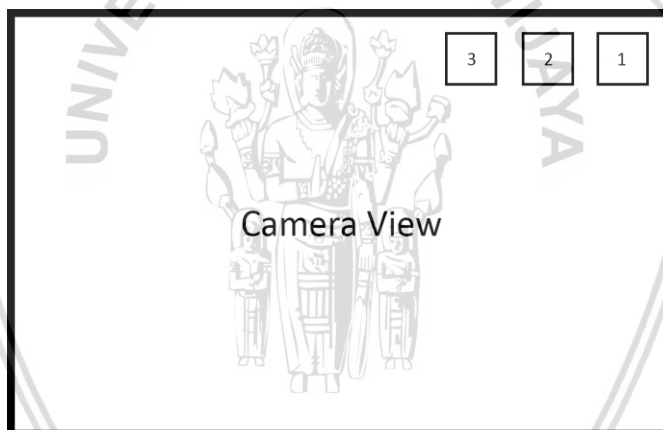
Gambar 4.11 Perancangan halaman awal / menu utama

Gambar 4.11 merupakan perancangan dari menu utama / halaman awal ketika *user* membuka aplikasi. Menu utama terdiri dari beberapa tombol yaitu tombol mulai, bantuan, tentang, dan keluar. Tabel 4.13 berisi penjelasan dari masing-masing tombol.

Tabel 4.13 Penjelasan tombol pada halaman menu utama

Nama	Penjelasan
Mulai	Tombol ini akan mengarahkan <i>user</i> ke halaman <i>camera view</i> yang digunakan untuk mengidentifikasi <i>marker</i> .
Bantuan	Tombol ini akan mengarahkan <i>user</i> ke halaman yang berisi panduan atau langkah-langkah dalam mengoperasikan aplikasi.
Tentang	Tombol ini akan mengarahkan <i>user</i> ke halaman yang berisi informasi tentang aplikasi.
Keluar	Tombol ini merupakan tombol untuk keluar dari aplikasi.

b. Tampilan *camera view*



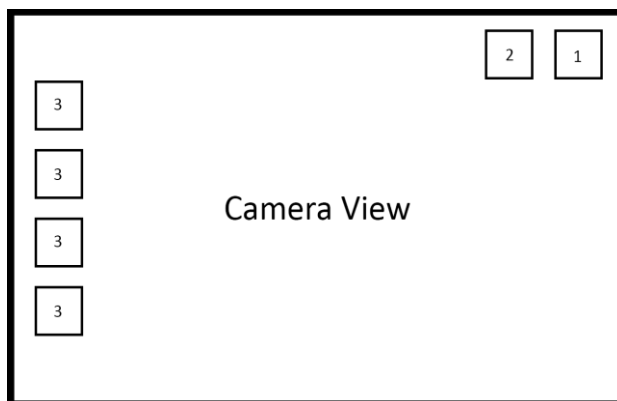
Gambar 4.12 Perancangan halaman pada saat *marker* teridentifikasi valid

Gambar 4.12 merupakan perancangan dari halaman *camera view* setelah *user* menekan tombol “mulai” dan *marker* teridentifikasi valid. Pada halaman ini terdiri dari beberapa tombol yaitu tombol untuk menampilkan varian huruf, tombol untuk menampilkan transliterasi, dan tombol untuk kembali ke menu utama / halaman awal. Tabel 4.14 berisi penjelasan dari masing-masing tombol.

Tabel 4.14 Penjelasan tombol pada saat *marker* teridentifikasi valid

Nomor	Penjelasan
1	Tombol ini digunakan kembali ke menu utama / halaman awal.
2	Tombol ini digunakan untuk menampilkan transliterasi huruf.
3	Tombol ini digunakan untuk menampilkan varian huruf.

c. Tampilan halaman untuk menampilkan varian dan transliterasi



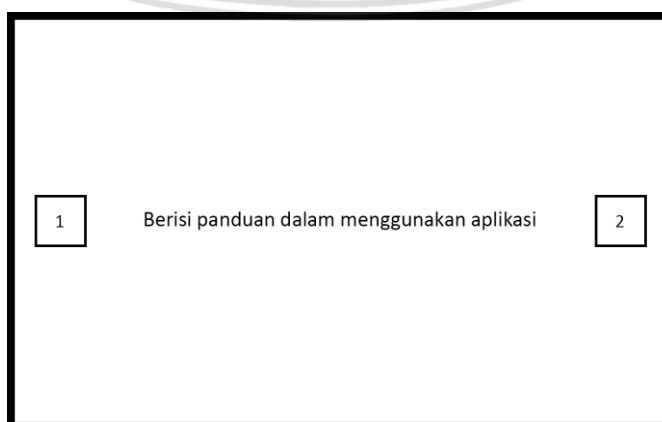
Gambar 4.13 Perancangan halaman pada fitur varian dan transliterasi

Gambar 4.13 merupakan perancangan dari halaman *camera view* setelah *user* menekan tombol untuk melihat varian huruf. Pada halaman ini terdiri dari beberapa tombol yaitu tombol untuk melihat varian-varian huruf, tombol untuk menampilkan animasi penulisan huruf, dan tombol untuk kembali ke menu utama atau halaman awal. Tabel 4.15 berisi penjelasan dari masing-masing tombol.

Tabel 4.15 Penjelasan tombol pada fitur animasi dan suara

Nomor	Penjelasan
1	Tombol ini digunakan kembali ke menu utama / halaman awal.
2	Tombol ini digunakan untuk menampilkan animasi atau memutar suara pembacaan huruf.
3	Tombol ini digunakan untuk menampilkan varian-varian huruf atau transliterasi lainnya.

d. Tampilan halaman bantuan



Gambar 4.14 Perancangan halaman bantuan

Gambar 4.14 merupakan halaman yang menampilkan informasi tentang cara menggunakan aplikasi sehingga *user* dapat menggunakan fitur-fitur yang ada di dalam aplikasi dengan benar. Tabel 4.16 berisi penjelasan dari tombol-tombol yang ada pada halaman bantuan.

Tabel 4.16 Penjelasan tombol pada halaman bantuan

Nomor	Penjelasan
1	Tombol ini digunakan untuk menampilkan informasi yang selanjutnya.
2	Tombol ini digunakan untuk menampilkan informasi yang sebelumnya.

e. Tampilan halaman tentang

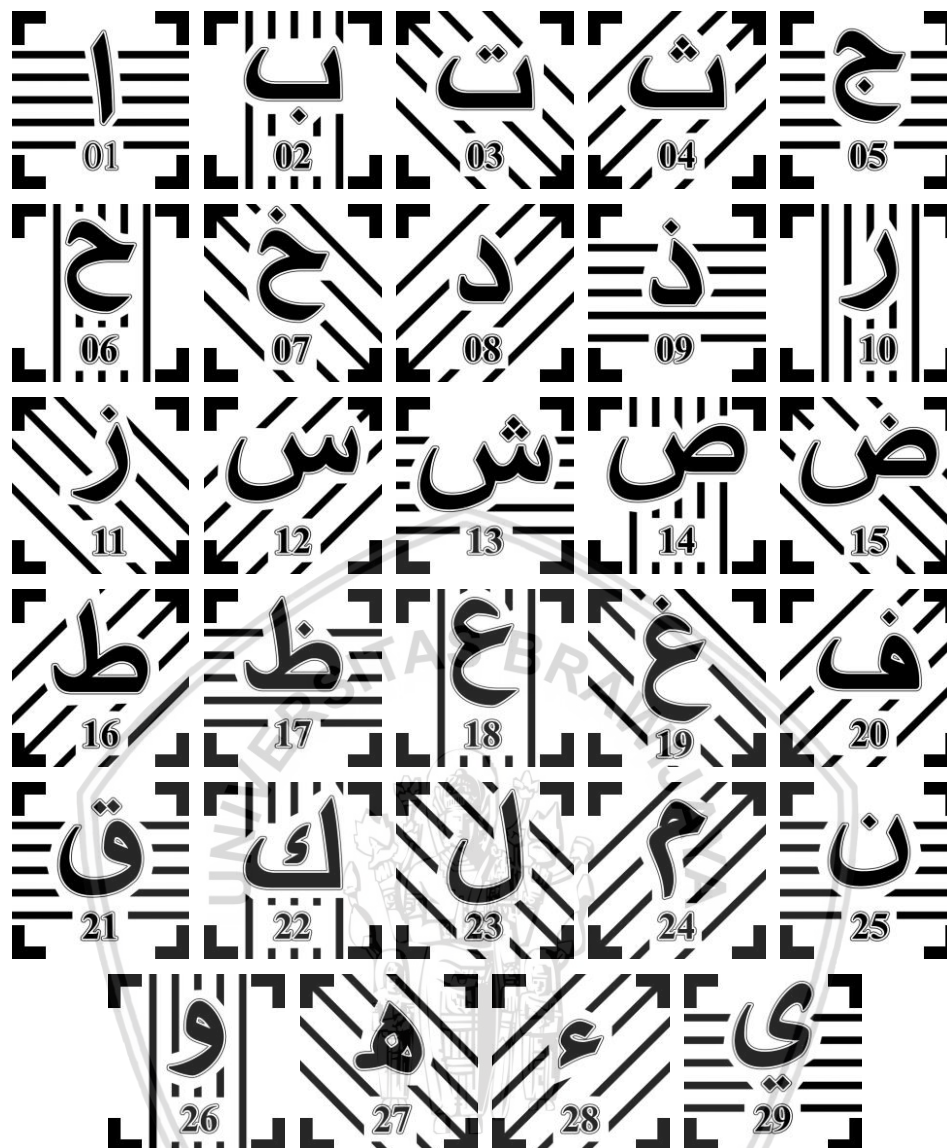


Gambar 4.15 Perancangan halaman tentang

Gambar 4.15 merupakan perancangan halaman yang nantinya akan menampilkan informasi tentang aplikasi.

4.3.5 Perancangan *marker*

Desain *marker* dibuat menggunakan aplikasi pengolah grafis photoshop. *Marker* yang dibuat adalah gambar berbentuk persegi dengan salah satu huruf hijaiyah di tengahnya. Pemberian nomor dan motif pada *marker* dilakukan untuk menghindari terjadinya kekeliruan pada aplikasi saat mendeteksi *marker*, seperti pada saat mendeteksi *marker* dengan huruf “Ta” yang mirip dengan huruf “Ša” atau huruf “Ha” yang mirip dengan huruf “Jim” atau “Kha”.



Gambar 4.16 Gambar desain *marker*

Gambar 4.16 adalah 29 gambar desain yang akan digunakan sebagai *marker* untuk menampilkan objek 3D pada aplikasi. *Marker* yang akan dibuat adalah *marker* berbentuk persegi dengan ukuran 2,5x2,5cm; 5x5cm; 7,5x7,5cm; 10x10cm; dan 12,5x12,5cm.

BAB 5 IMPLEMENTASI

Pada bab ini akan dibahas tentang tahapan dan implementasi aplikasi sesuai dengan hasil analisis kebutuhan dan perancangan aplikasi. Pembahasan terdiri dari penjelasan mengenai spesifikasi lingkungan aplikasi, implementasi *marker*, implementasi konten 3D, implementasi antarmuka aplikasi, dan implementasi kode program.

5.1 Spesifikasi Lingkungan Implementasi

Lingkungan implementasi dibagi menjadi dua yaitu spesifikasi perangkat keras dan perangkat lunak.

5.1.1 Spesifikasi perangkat keras

Spesifikasi perangkat keras yang digunakan dalam proses pengembangan aplikasi dijelaskan pada Tabel 5.1.

Tabel 5.1 Spesifikasi perangkat keras komputer

Nama Komponen	Spesifikasi
<i>Model</i>	Lenovo Ideapad Z410
<i>Processor</i>	Intel Core i7 2.2 GHz 4702MQ
<i>Memory (RAM)</i>	4 GB
<i>Harddisk</i>	1 TB
<i>Graphic</i>	NVIDIA GeForce GT 740M 2GB
<i>Resolution</i>	1366 x 768

Sedangkan untuk implementasi pada perangkat *mobile* menggunakan spesifikasi perangkat keras yang dijelaskan pada Tabel 5.2.

Tabel 5.2 Spesifikasi perangkat keras *smartphone*

Nama Komponen	Spesifikasi
<i>Model</i>	Asus Zenfone Selfie ZD551KL
<i>Processor</i>	Octa-core (4x1.7 GHz Cortex-A53 & 4x1.0 GHz Cortex-A53)
<i>Memory (RAM)</i>	3 GB
<i>Camera</i>	13 MP
<i>Graphic</i>	Adreno 405
<i>Resolution</i>	1080 x 1920

5.1.2 Spesifikasi perangkat lunak

Spesifikasi perangkat lunak yang digunakan dalam proses pengembangan aplikasi dijelaskan pada Tabel 5.3.

Tabel 5.3 Spesifikasi perangkat lunak komputer

Nama Komponen	Spesifikasi
<i>Operating System</i>	Windows 10 Home Single Language 64-bit
<i>3D Object Editor</i>	Blender 2.78c
<i>Image Editor</i>	Adobe Photoshop CC 2017
<i>Engine</i>	Unity 5.5.4p4

Sedangkan untuk implementasi pada perangkat *mobile* menggunakan spesifikasi perangkat lunak yang dijelaskan pada Tabel 5.4.

Tabel 5.4 Spesifikasi perangkat lunak *smartphone*

Nama Komponen	Spesifikasi
<i>Operating System</i>	Android 5.0, dan 6.0

5.2 Implementasi *Marker*

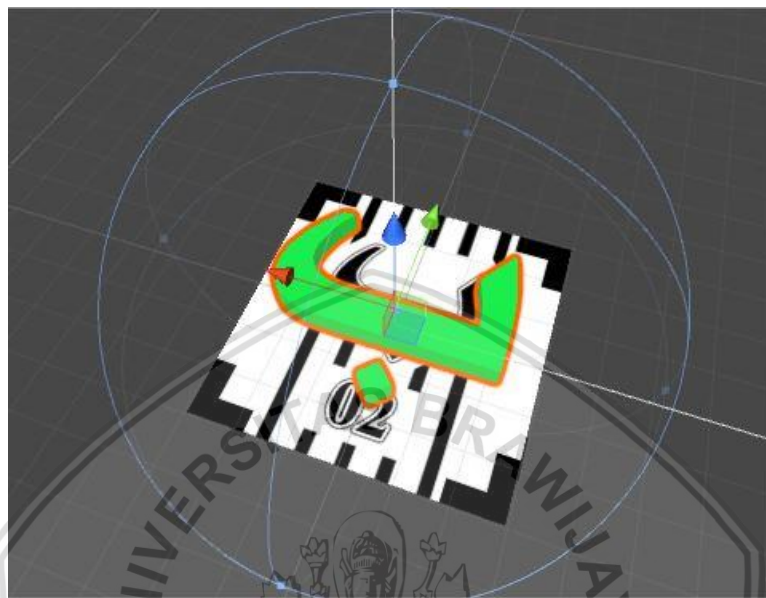
Pada implementasi *marker*, gambar yang sudah dibuat akan dimasukkan ke dalam *database* vuforia yang nantinya akan diberikan titik-titik penanda sebagai acuan untuk mengidentifikasi *marker*. Gambar 5.1 menunjukkan gambar *marker* yang sudah diberikan titik-titik penanda oleh vuforia.



Gambar 5.1 *Marker* dengan tanda pengenalan

5.3 Implementasi Konten 3D

Pada tahap implementasi konten 3D, objek 3D yang sudah dibuat dengan aplikasi blender, dimasukkan ke dalam unity dan diatur ukuran serta penempatannya sesuai dengan *marker* yang telah ditentukan. Gambar 5.2 menunjukkan implementasi salah satu contoh objek 3D pada unity.



Gambar 5.2 Implementasi objek 3D

5.4 Implementasi Antarmuka Aplikasi

Pada bagian ini dijelaskan mengenai implementasi dari antarmuka aplikasi yaitu tampilan menu utama, tampilan *camera view*, tampilan halaman bantuan, serta tampilan halaman tentang.

5.4.1 Menu utama

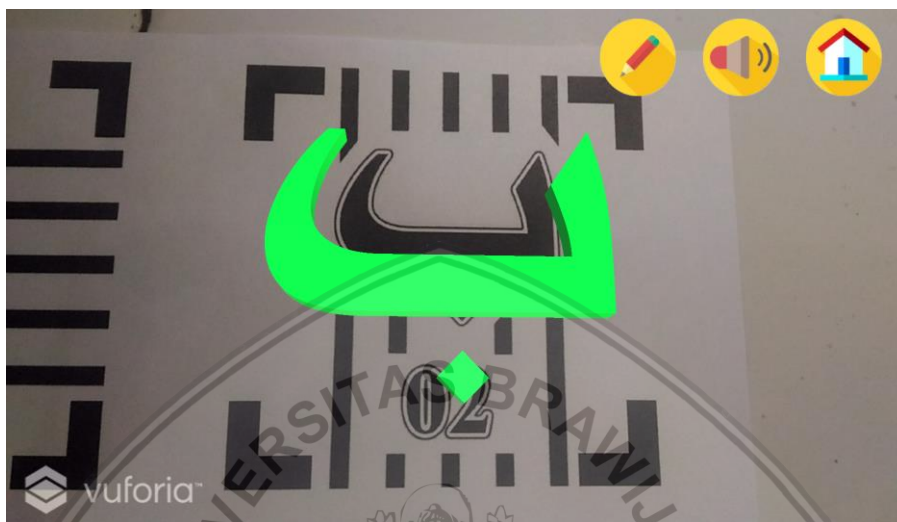


Gambar 5.3 Implementasi menu utama

Gambar 5.3 menampilkan implementasi tampilan menu utama dari aplikasi. Menu utaman ini menampilkan beberapa menu yaitu menu “Mulai”, menu “Bantuan”, menu “Tentang”, dan menu “Keluar”.

5.4.2 Camera view

5.4.2.1 Tampilan saat *marker* teridentifikasi valid



Gambar 5.4 Tampilan saat *marker* valid

Gambar 5.4 merupakan tampilan pada saat *marker* teridentifikasi valid yang menampilkan objek 3D dan fitur utama dari aplikasi. Tombol dengan gambar pensil merupakan tombol untuk melihat varian huruf, tombol dengan gambar *speaker* merupakan tombol untuk melihat tranliterasi, dan tombol dengan gambar rumah merupakan tombol untuk kembali ke menu utama.

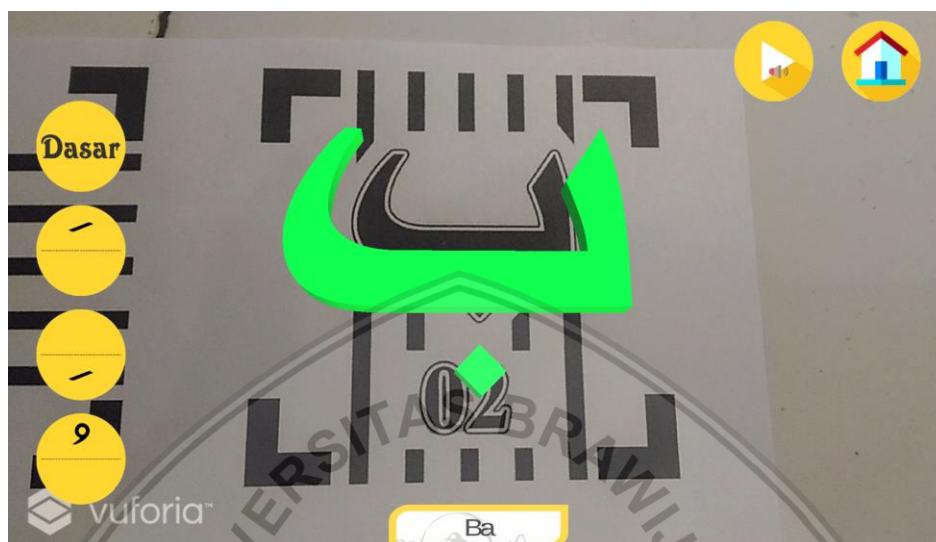
5.4.2.2 Tampilan saat memilih fitur varian huruf



Gambar 5.5 Tampilan saat memilih fitur varian

Gambar 5.5 merupakan tampilan saat memilih fitur varian. Tombol dengan tulisan dasar, awal, tengah dan akhir merupakan tombol untuk menampilkan varian dari huruf hijaiyah, sedangkan tombol dengan gambar segitiga putih dan pensil merupakan tombol untuk melihat animasi penulisan huruf.

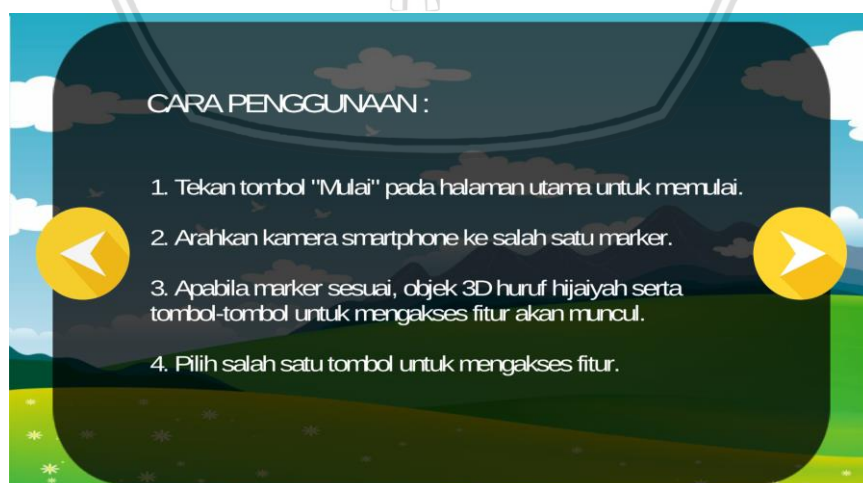
5.4.2.3 Tampilan saat memilih fitur transliterasi



Gambar 5.6 Tampilan saat memilih fitur transliterasi

Gambar 5.6 merupakan tampilan saat memilih fitur transliterasi. Tombol dengan tulisan dasar, *fatḥah*, *kasrah*, dan *dammah* merupakan tombol untuk melihat transliterasi dari huruf dasar dan huruf dengan *harakat*, sedangkan tombol dengan gambar segitiga putih dan *speaker* merupakan tombol untuk memutar suara pengucapan huruf.

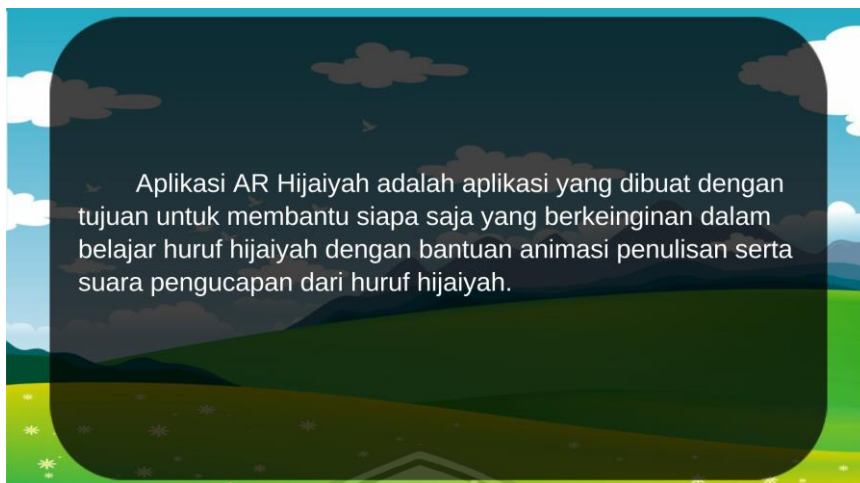
5.4.3 Halaman bantuan



Gambar 5.7 Implementasi halaman bantuan

Gambar 5.7 merupakan tampilan halaman bantuan. Pada halaman ini akan di jelaskan cara penggunaan aplikasi dan penjelasan tombol-tombol yang ada.

5.4.4 Halaman tentang



Gambar 5.8 Implementasi halaman tentang

Gambar 5.8 merupakan tampilan dari halaman tentang yang akan menampilkan informasi tentang aplikasi.

5.5 Implementasi Kode Program

Implementasi kode program adalah tahap penerapan perancangan dengan cara mengimplementasikannya dalam bentuk *pseudocode* sehingga menjadi sebuah sistem.

5.5.1 Implementasi deteksi *marker*

Pada Pseudocode 5.1 terdapat *method* *OnTrackingFound* yang menampung komponen-komponen *renderer* dan *collider* serta mengaktifkannya apabila *marker* terdeteksi.

Pseudocode 5.1 Implementasi deteksi *marker*

```
void OnTrackingFound {
    Renderer[] rendererComponents ← GetComponentsInChildren<Renderer>(true)
    Collider[] colliderComponents ← GetComponentsInChildren<Collider>(true)

    foreach (Renderer component in rendererComponents)
        set component.enabled to true

    foreach (Collider component in colliderComponents)
        set component.enabled to true
}
```

5.5.2 Implementasi fitur varian

Pada Pseudocode 5.2 terdapat *method* showDasar yang berfungsi untuk menampilkan bentuk dasar huruf, *method* showAwal untuk menampilkan bentuk awal huruf, *method* showTengah untuk menampilkan bentuk tengah huruf, dan *method* showAkhir untuk menampilkan bentuk akhir huruf.

Pseudocode 5.2 Implementasi fitur varian

```
void showDasar {  
    if varianAwal = true  
        set varianAwal to false  
    end if  
    if varianTengah = true  
        set varianTengah to false  
    end if  
    if varianAkhir = true  
        set varianAkhir to false  
    end if  
    change model to dasar  
    set varianDasar to true  
}  
  
void showAwal {  
    if varianDasar = true  
        set varianDasar to false  
    end if  
    if varianTengah = true  
        set varianTengah to false  
    end if  
    if varianAkhir = true  
        set varianAkhir to false  
    end if  
    change model to awal  
    set varianAwal to true  
}  
  
void showTengah {  
    if varianDasar = true  
        set varianDasar to false  
    end if  
    if varianAwal = true  
        set varianAwal to false  
    end if  
    if varianAkhir = true  
        set varianAkhir to false
```

Pseudocode 5.2 Implementasi fitur varian (lanjutan)

```

    end if
    change model to tengah
    set varianTengah to true
}

void showAkhir {
    if varianDasar = true
        set varianDasar to false
    end if
    if varianAwal = true
        set varianAwal to false
    end if
    if varianTengah = true
        set varianTengah to false
    end if
    change model to akhir
    set varianAkhir to true
}

```

5.5.3 Implementasi fitur animasi

Pada Pseudocode 5.3 terdapat *method* Write yang akan memberikan nilai variabel destination dan slider. Variabel tersebut digunakan pada *method* Update yang akan merubah nilai *_Cutoff* alpha secara perlahan.

Pseudocode 5.3 Implementasi fitur animasi

```

void Update {
    slider ← interpolasi dari nilai slider dan destination
    material.SetFloat ("_Cutoff",slider)
}

static void Write {
    destination ← 0;
    slider ← 1;
}

```

5.5.4 Implementasi fitur transliterasi

Pada Pseudocode 5.4 terdapat *method* showDasar2 yang berfungsi untuk menampilkan transliterasi dasar huruf serta menonaktifkan objek *fathah*, *kasrah*, atau *dammah* apabila aktif. *Method* showFathah berfungsi untuk menampilkan transliterasi huruf dengan *harakat fathah* serta menonaktifkan objek *kasrah* atau *dammah*. *Method* showKasrah berfungsi untuk menampilkan transliterasi huruf

dengan *harakat kasrah* serta menonaktifkan objek *fathah* atau *dammah*. Dan *method* showDammah berfungsi untuk menampilkan transliterasi huruf dengan harakat *dammah* serta menonaktifkan objek *fathah* atau *kasrah*.

Pseudocode 5.4 Implementasi fitur transliterasi

```
void showDasar2 {
    if fathah = true
        set modelFathah.SetActive to false
        set fathah to false
    end if
    if kasrah = true
        modelKasrah.SetActive to false
        set kasrah to false
    end if
    if dammah = true
        modelDammah.SetActive to false
        set dammah to false
    end if
    change translation to dasar
    set dasar2 to true
}
```

```
void showFathah {
    if dasar2 = true
        set modelDasar2.SetActive to false
        set dasar2 to false
    end if
    if kasrah = true
        modelKasrah.SetActive to false
        set kasrah to false
    end if
    if dammah = true
        modelDammah.SetActive to false
        set dammah to false
    end if
    change translation to fathah
    set fathah to true
}
```

```
void showKasrah {
    if dasar2 = true
        set modelDasar2.SetActive to false
        set dasar2 to false
    end if
    if fathah = true
```

Pseudocode 5.4 Implementasi fitur transliterasi (lanjutan)

```

        modelFathah.SetActive to false
        set fathah to false
    end if
    if dammah = true
        modelDammah.SetActive to false
        set dammah to false
    end if
    change translation to kasrah
    set kasrah to true
}

void showDammah {
    if dasar2 = true
        set modelDasar2.SetActive to false
        set dasar2 to false
    end if
    if fathah = true
        modelFathah.SetActive to false
        set fathah to false
    end if
    if kasrah = true
        modelKasrah.SetActive to false
        set kasrah to false
    end if
    change translation to dammah
    set dammah to true
}

```

5.5.5 Implementasi fitur pemutaran suara

Pada Pseudocode 5.5 terdapat *method* playSound yang berfungsi untuk memutar suara pengucapan huruf berdasarkan variabel yang sedang aktif. Apabila variabel yang sedang aktif adalah variabel dasar, maka suara pengucapan huruf yang akan diputar adalah nama huruf tersebut. Jika variabel yang sedang aktif adalah variabel fathah, maka suara yang akan diputar adalah suara pengucapan huruf dengan harakat *fathah*. Jika variabel yang sedang aktif adalah variabel kasrah, maka suara yang akan diputar adalah suara pengucapan huruf dengan harakat kasrah. Dan jika variabel yang sedang aktif adalah variabel dammah, maka suara yang akan diputar adalah suara pengucapan huruf dengan *harakat dammah*.

Pseudocode 5.5 Implementasi fitur pemutaran suara

```
void playSound {  
    if dasar2 = true  
        play dasar2Sound  
    end if  
    if fathah = true  
        play fathahSound  
    end if  
    if kasrah = true  
        play kasrahSound  
    end if  
    if dammah = true  
        play dammahSound  
    end if  
}
```



BAB 6 PENGUJIAN DAN ANALISIS

Pada bab ini dibahas tentang pengujian aplikasi serta analisis dari hasil pengujian yang telah dilakukan. Pengujian yang dilakukan menggunakan metode *blackbox testing*. Analisis pengujian dilakukan setelah tahap pengujian selesai dan nantinya akan digunakan dalam penarikan kesimpulan.

6.1 Pengujian

Pengujian yang akan dilakukan diantaranya adalah pengujian validasi, jarak optimal *marker*, dan pengujian aplikasi.

6.1.1 Pengujian validasi

Pengujian validasi dilakukan untuk mengetahui apakah aplikasi yang dibuat berjalan sesuai dengan skenario *use case* yang telah dibuat pada saat tahap perancangan. Karena pengujian ini hanya berfokus untuk mengetahui apakah aplikasi berjalan sesuai dengan skenario dan kebutuhan, maka metode pengujian yang digunakan adalah *blackbox testing*.

Tabel 6.1 Kasus uji melihat halaman bantuan

Nama Kasus Uji	Melihat halaman bantuan
Objek Uji	Kebutuhan Fungsional (SRS_000_01)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk menampilkan halaman bantuan.
Prosedur Uji	1. Membuka aplikasi. 2. Menekan tombol menu “Bantuan”.
Hasil yang Diharapkan	Aplikasi dapat menampilkan halaman bantuan.

Tabel 6.2 Kasus uji melihat halaman tentang

Nama Kasus Uji	Melihat halaman tentang
Objek Uji	Kebutuhan Fungsional (SRS_000_02)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk menampilkan halaman tentang.
Prosedur Uji	1. Membuka aplikasi. 2. Menekan tombol menu “Tentang”.
Hasil yang Diharapkan	Aplikasi dapat menampilkan halaman tentang.

Tabel 6.3 Kasus uji keluar aplikasi

Nama Kasus Uji	Keluar aplikasi
Objek Uji	Kebutuhan Fungsional (SRS_000_03)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk menutup aplikasi.
Prosedur Uji	<ol style="list-style-type: none"> 1. Membuka aplikasi. 2. Menekan tombol menu “Keluar”. 3. Menekan tombol “Ya”.
Hasil yang Diharapkan	Aplikasi dapat menutup aplikasi.

Tabel 6.4 Kasus uji memulai mode AR

Nama Kasus Uji	Memulai mode AR
Objek Uji	Kebutuhan Fungsional (SRS_001_01)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk mengidentifikasi <i>marker</i> .
Prosedur Uji	<ol style="list-style-type: none"> 1. Membuka aplikasi. 2. Menekan tombol menu “Mulai”. 3. Mengarahkan kamera pada <i>marker</i>.
Hasil yang Diharapkan	Aplikasi dapat masuk pada mode <i>camera view</i> , melakukan pengidentifikasian <i>marker</i> , serta menampilkan objek 3D dan mengaktifkan fitur yang ada.

Tabel 6.5 Kasus uji lihat varian huruf

Nama Kasus Uji	Lihat varian huruf
Objek Uji	Kebutuhan Fungsional (SRS_002_01)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk menampilkan varian huruf.

Tabel 6.5 Kasus uji lihat varian huruf (lanjutan)

Prosedur Uji	<ol style="list-style-type: none"> 1. Membuka aplikasi. 2. Menekan tombol menu “Mulai”. 3. Mengarahkan kamera pada <i>marker</i>. 4. Menekan tombol fitur varian.
Hasil yang Diharapkan	Aplikasi dapat menampilkan varian huruf dari masing-masing <i>marker</i> .

Tabel 6.6 Kasus uji lihat animasi penulisan

Nama Kasus Uji	Lihat animasi penulisan
Objek Uji	Kebutuhan Fungsional (SRS_002_02)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk menampilkan animasi penulisan huruf.
Prosedur Uji	<ol style="list-style-type: none"> 1. Membuka aplikasi. 2. Menekan tombol menu “Mulai”. 3. Mengarahkan kamera pada <i>marker</i>. 4. Menekan tombol fitur varian. 5. Menekan tombol fitur animasi.
Hasil yang Diharapkan	Aplikasi dapat menampilkan animasi penulisan huruf dari masing-masing <i>marker</i> .

Tabel 6.7 Kasus uji lihat transliterasi

Nama Kasus Uji	Lihat transliterasi
Objek Uji	Kebutuhan Fungsional (SRS_003_01)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk menampilkan transliterasi dari huruf.
Prosedur Uji	<ol style="list-style-type: none"> 1. Membuka aplikasi. 2. Menekan tombol menu “Mulai”. 3. Mengarahkan kamera pada <i>marker</i>. 4. Menekan tombol fitur transliterasi.
Hasil yang Diharapkan	Aplikasi dapat menampilkan transliterasi huruf dari masing-masing <i>marker</i> .

Tabel 6.8 Kasus uji dengar pembacaan huruf

Nama Kasus Uji	Dengar pembacaan huruf
Objek Uji	Kebutuhan Fungsional (SRS_003_02)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk memutar suara pengucapan huruf.
Prosedur Uji	<ol style="list-style-type: none"> 1. Membuka aplikasi. 2. Menekan tombol menu “Mulai”. 3. Mengarahkan kamera pada <i>marker</i>. 4. Menekan tombol fitur transliterasi. 5. Menekan tombol fitur suara pengucapan.
Hasil yang Diharapkan	Aplikasi dapat memutar suara pengucapan huruf dari masing-masing <i>marker</i> .

6.1.2 Pengujian kondisi optimal *marker*

Pengujian kondisi optimal *marker* dicari berdasarkan besar dan jarak yang dibutuhkan aplikasi untuk mengenali *marker* dan menampilkan objek 3D secara utuh beserta fiturnya. Besar ukuran *marker* yang diuji adalah 2,5x2,5 cm; 5x5 cm; 7,5x7,5 cm; 10x10 cm; dan 12,5x12,5 cm. Sedangkan jarak yang diuji mulai dari 4cm-45cm dengan selisih 1cm secara tegak lurus dari *marker*.

6.1.3 Pengujian aplikasi

Pengujian aplikasi dilakukan dengan melihat peningkatan nilai membaca siswa. Pengujian dilakukan pada siswa kelas I SD Khoiru Ummah Kranggan kota Bekasi yang berjumlah 20 anak dan dengan bantuan dari guru yang mengajar untuk menggunakan aplikasi.

6.2 Hasil dan Analisis

Berdasarkan kasus yang diujikan, berikut ini adalah hasil dan analisis yang diperoleh dari pengujian sesuai dengan kasus ujinya.

6.2.1 Hasil dan analisis pengujian validasi

Tabel 6.9 Hasil pengujian validasi

No.	Nama Kasus Uji	Hasil yang Didapatkan	Status Validitas
1.	Melihat halaman bantuan (SRS_000_01)	Aplikasi dapat menampilkan halaman bantuan.	Valid

Tabel 6.9 Hasil pengujian validasi (lanjutan)

2.	Melihat halaman tentang (SRS_000_02)	Aplikasi dapat menampilkan halaman tentang.	Valid
3.	Keluar aplikasi (SRS_000_03)	Aplikasi dapat menutup aplikasi.	Valid
4.	Memulai mode AR (SRS_001_01)	Aplikasi dapat masuk pada mode <i>camera view</i> , melakukan pengidentifikasian <i>marker</i> , serta menampilkan objek 3D dan mengaktifkan fitur yang ada.	Valid
5.	Lihat varian huruf (SRS_002_01)	Aplikasi dapat menampilkan varian huruf dari masing-masing <i>marker</i> .	Valid
6.	Lihat animasi penulisan (SRS_002_02)	Aplikasi dapat menampilkan animasi penulisan huruf dari masing-masing <i>marker</i> .	Valid
7.	Lihat transliterasi (SRS_003_01)	Aplikasi dapat menampilkan transliterasi huruf dari masing-masing <i>marker</i> .	Valid
8.	Dengar pembacaan huruf (SRS_003_02)	Aplikasi dapat menampilkan transliterasi huruf dari masing-masing <i>marker</i> .	Valid

Analisis dari hasil pengujian validasi bertujuan melihat kesesuaian antara kebutuhan fungsional dan hasil kinerja sistem. Berdasarkan Tabel 6.9 dapat disimpulkan bahwa fungsionalitas serta implementasi dari aplikasi yang dibuat sudah memenuhi kebutuhan fungsional yang dijabarkan pada tahap analisis kebutuhan.

6.2.2 Hasil dan analisis pengujian kondisi optimal *marker*

Tabel 6.10 Hasil pengujian kondisi optimal *marker*

Jarak(cm) \ Besar(cm)	2,5x2,5	5x5	7,5x7,5	10x10	12,5x12,5
4	TRUE	FALSE	FALSE	FALSE	FALSE
5	TRUE	FALSE	FALSE	FALSE	FALSE
6	TRUE	FALSE	FALSE	FALSE	FALSE
7	TRUE	FALSE	FALSE	FALSE	FALSE
8	TRUE	TRUE	FALSE	FALSE	FALSE
9	FALSE	TRUE	FALSE	FALSE	FALSE

Tabel 6.10 Hasil pengujian kondisi optimal *marker* (lanjutan)

10	FALSE	TRUE	FALSE	FALSE	FALSE
11	FALSE	TRUE	FALSE	FALSE	FALSE
12	FALSE	TRUE	TRUE	FALSE	FALSE
13	FALSE	TRUE	TRUE	FALSE	FALSE
14	FALSE	TRUE	TRUE	FALSE	FALSE
15	FALSE	TRUE	TRUE	FALSE	FALSE
16	FALSE	TRUE	TRUE	TRUE	FALSE
17	FALSE	TRUE	TRUE	TRUE	FALSE
18	FALSE	FALSE	TRUE	TRUE	FALSE
19	FALSE	FALSE	TRUE	TRUE	TRUE
20	FALSE	FALSE	TRUE	TRUE	TRUE
21	FALSE	FALSE	TRUE	TRUE	TRUE
22	FALSE	FALSE	TRUE	TRUE	TRUE
23	FALSE	FALSE	TRUE	TRUE	TRUE
24	FALSE	FALSE	TRUE	TRUE	TRUE
25	FALSE	FALSE	TRUE	TRUE	TRUE
26	FALSE	FALSE	TRUE	TRUE	TRUE
27	FALSE	FALSE	FALSE	TRUE	TRUE
28	FALSE	FALSE	FALSE	TRUE	TRUE
29	FALSE	FALSE	FALSE	TRUE	TRUE
30	FALSE	FALSE	FALSE	TRUE	TRUE
31	FALSE	FALSE	FALSE	TRUE	TRUE
32	FALSE	FALSE	FALSE	TRUE	TRUE
33	FALSE	FALSE	FALSE	TRUE	TRUE
34	FALSE	FALSE	FALSE	TRUE	TRUE
35	FALSE	FALSE	FALSE	FALSE	TRUE
36	FALSE	FALSE	FALSE	FALSE	TRUE
37	FALSE	FALSE	FALSE	FALSE	TRUE
38	FALSE	FALSE	FALSE	FALSE	TRUE
39	FALSE	FALSE	FALSE	FALSE	TRUE
40	FALSE	FALSE	FALSE	FALSE	TRUE
41	FALSE	FALSE	FALSE	FALSE	TRUE
42	FALSE	FALSE	FALSE	FALSE	TRUE
43	FALSE	FALSE	FALSE	FALSE	FALSE
44	FALSE	FALSE	FALSE	FALSE	FALSE
45	FALSE	FALSE	FALSE	FALSE	FALSE

Tabel 6.10 menunjukkan hasil dari pengujian kondisi optimal *marker* berdasarkan besar dan jarak. Keterangan *TRUE* menandakan bahwa *marker* terdeteksi dan dapat menampilkan objek secara utuh, sedangkan keterangan

FALSE menandakan bahwa *marker* tidak terdeteksi atau *marker* terdeteksi namun objek tidak tampil atau tampil secara tidak utuh.

Berdasarkan Tabel 6.10 juga dapat dilihat bahwa kondisi optimal *marker* dengan ukuran 2,5x2,5cm adalah antara 4-8cm, *marker* dengan ukuran 5x5cm adalah antara 8-17cm, *marker* dengan ukuran 7,5x7,5cm adalah antara 12-26cm, *marker* dengan ukuran 10x10cm adalah antara 16-34cm, dan *marker* dengan ukuran 12,5x12,5cm adalah antara 19-42cm. Sehingga dapat disimpulkan bahwa semakin besar *marker*, maka akan semakin besar pula jarak untuk mendeteksinya.

6.2.3 Hasil dan analisis pengujian aplikasi

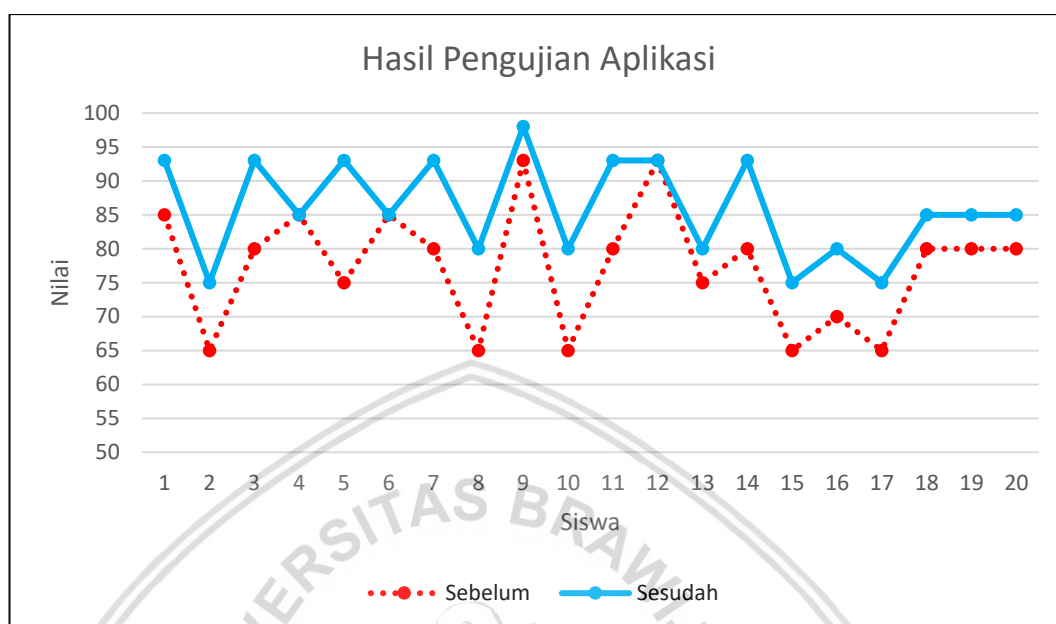
Hasil dari pengujian yang dilakukan dengan cara melihat peningkatan nilai membaca siswa dapat dilihat pada Tabel 6.11.

Tabel 6.11 Hasil pengujian aplikasi

No.	Nama	Nilai		
		Sebelum	Sesudah	Kenaikan
1	Siswa 1	85	93	9,41%
2	Siswa 2	65	75	15,38%
3	Siswa 3	80	93	16,25%
4	Siswa 4	85	85	0%
5	Siswa 5	75	93	24,00%
6	Siswa 6	85	85	0%
7	Siswa 7	80	93	16,25%
8	Siswa 8	65	80	23,08%
9	Siswa 9	93	98	5,38%
10	Siswa 10	65	80	23,08%
11	Siswa 11	80	93	16,25%
12	Siswa 12	93	93	0%
13	Siswa 13	75	80	6,67%
14	Siswa 14	80	93	16,25%
15	Siswa 15	65	75	15,38%
16	Siswa 16	70	80	14,29%
17	Siswa 17	65	75	15,38%
18	Siswa 18	80	85	6,25%
19	Siswa 19	80	85	6,25%
20	Siswa 20	80	85	6,25%
Rata-rata		77,3	85,95	11,19%

Berdasarkan Tabel 6.11 dapat disimpulkan bahwa aplikasi *mobile augmented reality* sebagai media belajar pengenalan dasar huruf hijaiyah meningkatkan nilai siswa sebesar 11,19% dari yang semula memiliki rata-rata nilai sebesar 77,3

sebelum menggunakan aplikasi menjadi 85,95 setelah menggunakan aplikasi. Gambar 6.1 merupakan grafik hasil pengujian aplikasi.



Gambar 6.1 Grafik hasil pengujian aplikasi

BAB 7 PENUTUP

7.1 Kesimpulan

Berdasarkan hasil tahapan sebelumnya mulai dari tahap perancangan, tahap implementasi, tahap pengujian, serta tahap analisis yang telah dilakukan, maka dapat diambil kesimpulan sebagai berikut:

1. Aplikasi yang dibuat menggunakan teknologi *augmented reality* sehingga dapat menampilkan objek 3D dari huruf hijaiyah serta menampilkan animasinya penulisannya. Antarmuka dan menu dibuat dengan sederhana, menggunakan warna yang cerah, dan menggunakan bahasa Indonesia sehingga mudah untuk digunakan.
2. Berdasarkan pengujian kondisi optimal *marker* yang berdasarkan pada besar dan jarak, *marker* dengan ukuran 2,5x2,5cm adalah antara 4-8cm, *marker* dengan ukuran 5x5cm adalah antara 8-17cm, *marker* dengan ukuran 7,5x7,5cm adalah antara 12-26cm, *marker* dengan ukuran 10x10cm adalah antara 16-34cm, dan *marker* dengan ukuran 12,5x12,5cm adalah antara 19-42cm.
3. Berdasarkan pengujian aplikasi yang dilakukan pada siswa kelas I SD Khoiru Ummah Kranggan kota Bekasi, aplikasi *mobile augmented reality* membantu dalam pembelajaran pengenalan huruf hijayah yang dapat dilihat dari meningkatnya rata-rata nilai siswa sebesar 11,19%.

7.2 Saran

Aplikasi *mobile augmented reality* ini masih perlu dikembangkan lagi agar menjadi lebih baik sehingga saran dibutuhkan untuk meningkatkan kualitas dari aplikasi. Saran yang diberikan untuk pengembang selanjutnya antara lain:

1. Meningkatkan kualitas animasi penulisan huruf hijaiyah agar lebih terlihat seperti tulisan tangan.
2. Mengembangkan fungsi yang dapat memberikan contoh penulisan huruf dalam suatu kata sehingga dapat memperjelas bentuk huruf apabila berada di awal, tengah, dan akhir.
3. Melakukan penelitian lebih lanjut tentang apa yang menyebabkan nilai anak tidak mengalami perubahan setelah menggunakan aplikasi.

DAFTAR PUSTAKA

- Apriyani, M. E., Huda, M. & Prasetyaningsih, S., 2016. Analisis Penggunaan Marker Tracking Pada Augmented Reality Huruf Hijaiyah. *Jurnal Infotel*, 8(1), pp. 71-77.
- Asai, K., Kobayashi, H. & Kondo, T., 2005. *Augmented Instructions - A Fusion of Augmented Reality and Printed Learning Materials* –. Kaohsiung, IEEE.
- Asyar, R., 2012. *Kreatif Mengembangkan Media Pembelajaran*. Jakarta: Gaung Persada Press.
- Azuma, R. T., 1997. A Survey of Augmented Reality. *Presence*, 6(4), pp. 355-385.
- Cushman, D. & Habbak, H. E., 2013. *Developing AR Games for iOS and Android*. Birmingham: Packt Publishing.
- Dewi, R., 2005. *Berbagai Masalah Anak Taman Kanak-Kanak*. Jakarta: Depdiknas.
- Fine, R., 2017. *Unity Blogs*. [Online] Tersedia di: <https://blogs.unity3d.com/2017/08/11/unityscripts-long-ride-off-into-the-sunset/> [Diakses 25 Agustus 2017].
- Gilakjani, A. P., 2012. Visual, Auditory, Kinaesthetic Learning Styles and Their Impacts on English Language Teaching. *Journal of Studies in Education*, 2(1), pp. 104-113.
- Ifka, I., 2013. *KESULITAN PELAFALAN HURUF HIJAIYAH YANG TIDAK TERDAPAT DI HURUF INDONESIA PADA MASYARAKAT SARADAN WONOGIRI*. [e-book]. Tersedia melalui: DIGILIB UNNES <<http://lib.unnes.ac.id/>> [Diakses 12 Agustus 2018].
- Katiyar, A., Kalra, K. & Garg, C., 2015. Marker Based Augmented Reality. *Advances in Computer Science and Information Technology (ACSIT)*, 2(5), pp. 441-445.
- Lampel, J., 2015. *The Beginner's Guide to Blender*. [e-book]. Tersedia melalui: BlenderHD <<https://www.blenderhd.com/>> [Diakses 2 September 2017].
- Milgram, P., Takemura, H., Utsumi, A. & Kishino, F., 1994. Augmented Reality: A class of displays on the reality-virtuality continuum. *Telemanipulator and Telepresence Technologies*, Volume 2351, pp. 282-292.
- Nugroho, A., 2010. *Rekayasa Perangkat Lunak Berbasis Objek dengan Metode USDP*. Yogyakarta: Penerbit ANDI.
- Rochmani, S., 2011. *PENGUNAAN MEDIA GAMBAR UNTUK MENINGKATKAN PRESTASI BELAJAR BAHASA MANDARIN SISWA KELAS V DI SD TRIPUSAKA SURAKARTA*. [e-book]. Tersedia melalui: UNS Institutional Repository <<https://eprints.uns.ac.id/>> [Diakses 14 Agustus 2017].
- Sidiq, Z., 2016. PEMANFAATAN MEDIA PEMBELAJARAN BERBASIS TEKNOLOGI. *Edutech*, 15(1), pp. 98-108.

Sommerville, I., 2011. *SOFTWARE ENGINEERING*. 9th ed. America: Addison-Wesley.

